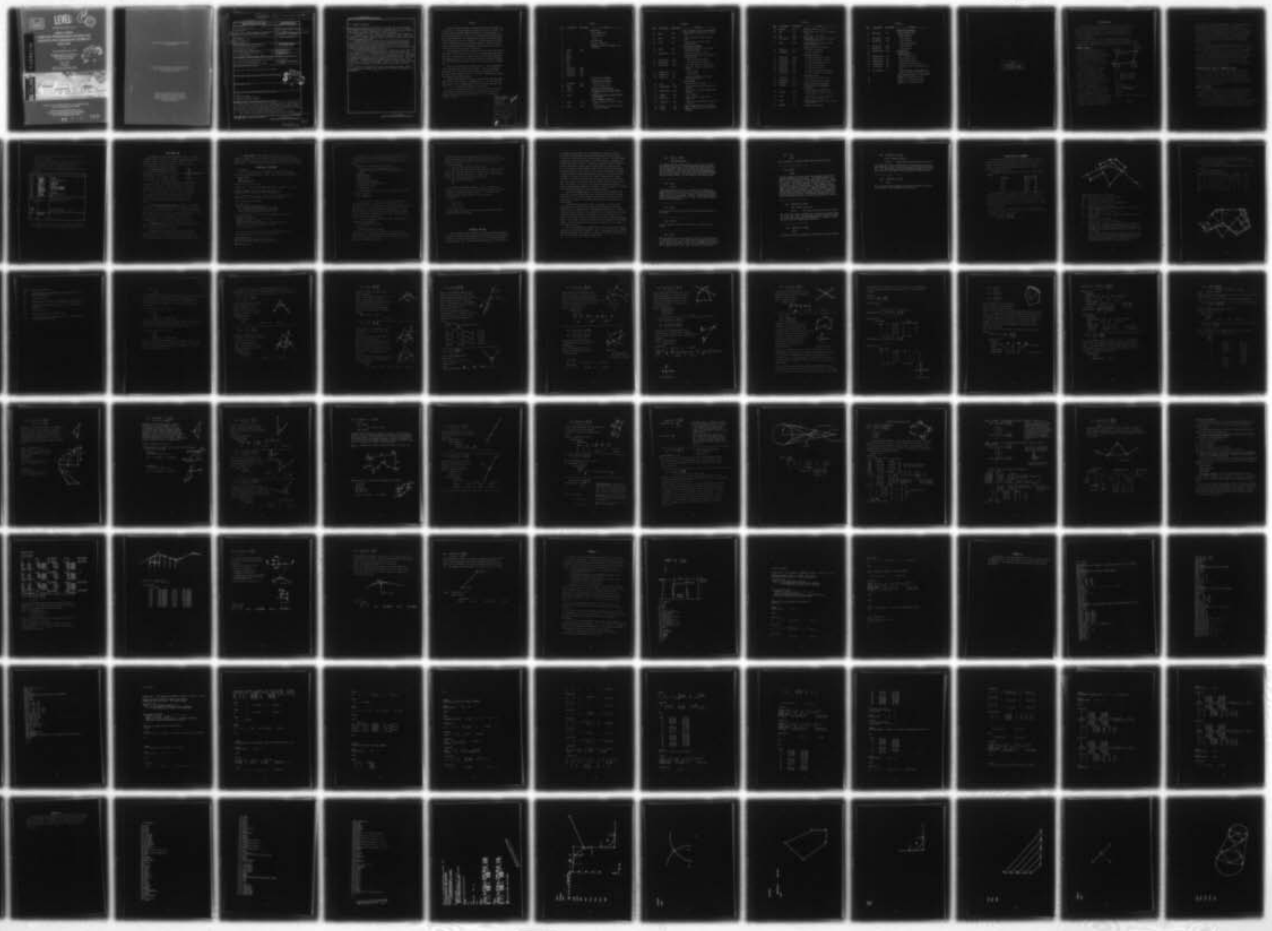
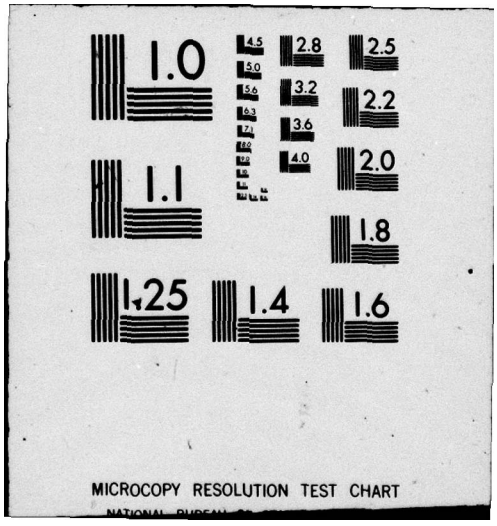
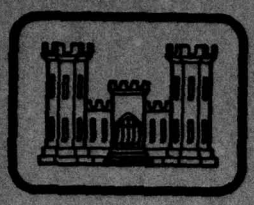
AD-A082 128 ARMY ENGINEER WATERWAYS EXPERIMENT STATION VICKSBURG MS F/G 9/2
USER'S GUIDE: COMPUTER PROGRAM WITH INTERACTIVE GRAPHICS FOR CO--ETC(U)
FEB 80 H B WILSON, J L HILL
UNCLASSIFIED WES INSTRUCTION-K-80-2 NL

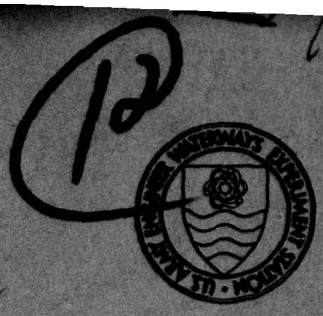
1 OF 2
ADA 082128




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1
8.0
9.0
MICROCOPY RESOLUTION TEST CHART



LEVEL II



INSTRUCTION REPORT K-80-2

USER'S GUIDE COMPUTER PROGRAM WITH INTERACTIVE GRAPHICS FOR COORDINATE GEOMETRY ANALYSIS

by

Howard B. Wilson, James L. Hill

Systems Engineering Consultants, Inc.
1900 Hackberry Lane, Suite B
Tuscaloosa, Ala. 35401

February 1980

Final Report

Destroy this report when no longer needed. Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

(14) WES INSTRUCTION—K-80-2

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: Instruction Report K-80-2
2. GOVT ACCESSION NO.:
3. RECIPIENT'S CATALOG NUMBER: (9)
4. TITLE (and Subtitle): (6) USER'S GUIDE: COMPUTER PROGRAM WITH INTERACTIVE GRAPHICS FOR COORDINATE GEOMETRY ANALYSIS.
5. TYPE OF REPORT & PERIOD COVERED: Final report. Jul 78 - Jul 79,
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): (10) Howard B. Wilson, James L. Hill
8. CONTRACT OR GRANT NUMBER(s): Dummy Source
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Systems Engineering Consultants, Inc., 1900 Hackberry Lane, Suite B, Tuscaloosa, Ala. 35401
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: (12) 105
11. CONTROLLING OFFICE NAME AND ADDRESS: (11) U. S. Army Engineer Division, Lower Mississippi Valley, P. O. Box 80, Vicksburg, Miss. 39180
12. REPORT DATE: February 1980
13. NUMBER OF PAGES: 100
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): U. S. Army Engineer Waterways Experiment Station, Automatic Data Processing Center, P. O. Box 631, Vicksburg, Miss. 39180
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:
16. DISTRIBUTION STATEMENT (of this Report): Approved for public release; distribution unlimited.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):
18. SUPPLEMENTARY NOTES:

DTIC ELECTE APR 1 1980 S D C

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

COGO
Computer programs
Coordinate geometry analysis
Interactive computer graphics

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report is a guide to use of a computer program for coordinate geometry analysis with interactive graphics, called "GCOGO." The original program has been revised to provide several additional commands plus graphic output for all commands for which such output would be helpful. The principal new features included in the version documented herein are as follows:

a. Azimuths can be measured clockwise from either north or south.
b. A feature allowing reuse of the immediately previous command is

(continued) → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

038100 DTIC

20. ABSTRACT (Continued)

provided. Most commands are followed by one or more lines of numeric data. When numeric data are entered in place of a command word, then it is assumed that use of the previous command word is implied and that the numeric data are appropriate input which would normally follow that command.

c. As many as 25 curves created by the ALIGNMENT or DEFCV commands can be stored and recalled at any time during an interactive session. Additional commands such as GETCRV, LSTCRV, and DLTCRV allow manipulation of these curves within the program work area.

d. GCOGO calculates in single precision rather than employing double precision as was done in the earlier version. This change was necessary because double precision variables would be incompatible with the Graphics Compatibility System (GCS) graphics software which is in single precision. Moreover, program storage limits did permit replication of the relevant graphics variables in both single and double precision.

e. The interactive graphics features are intended for use on a Tektronix 4014 terminal having an accompaning hardcopy device. A SHOW command with numeric options of 0, 1, or 2 allows the user to select (1) numeric output only, (2) graphical output only, or (3) both numeric and graphical output. The level of graphics output can be altered at any time during the analysis. Additional commands such as WINDOW, HRDCOPY, and ERASE also facilitate control of the screen area. Information on the screen is arranged such that numeric output is written on the left side with graphic results to the right. When the bottom of the screen is reached, the user must erase the screen and begin a new image.

f. Three new commands which locate points based on different triangle properties have been added. These commands are LOCATE/ANG2, LOCATE/ANG3, and LOCATE/DEF2.

PREFACE

This user's guide documents a computer program called GCOGO that can be used for coordinate geometry analysis with graphical output. The existing computer program, COGO, was modified to include graphics as part of the operation of the joint U. S. Army Engineer Waterways Experiment Station (WES) and U. S. Army Engineer Division, Lower Mississippi Valley, Computer Center for FY 1978-79. COGO was originally furnished to WES by Polycom Systems Ltd., Montreal, Canada, in 1969 and that version was modified for the WES Honeywell G-635 computer by Dr. Howard B. Wilson and Mr. William A. Price in 1975-1976.

The program modifications documented herein were coded for the WES G-635 by Dr. Wilson and Dr. James L. Hill of Systems Engineering Consultants, Inc., Tuscaloosa, Ala., under contract to WES during the period July 1978-July 1979. They also prepared this user's guide by revising the original COGO user's guide to incorporate the new graphics capabilities.

Work on the program was coordinated with U. S. Army Engineer Districts, Memphis (MD), and New Orleans (NOD). Liason was maintained between MD, NOD, and WES by means of telephone conversations with Ms. Wendy Truman, LMMED-DT, and Mr. Tom Phillips, LMNED-DD, who defined the desired program modifications.

Messrs. Price and Paul K. Senter of the Automatic Data Processing (ADP) Center, WES, had project responsibilities for the work under the supervision of Dr. N. Radhakrishnan, Special Technical Assistant, ADP Center. Mr. D. L. Neumann was Chief of the ADP Center.

COL J. L. Cannon, CE, and COL N. P. Conover, CE, were Directors of WES during the development of the program modifications and the preparation and publication of this report. Mr. F. R. Brown was Technical Director.

CONTENTS

CONTENTS

CONTENTS

CONTENTS

NOTE:
Throughout the text
of this user's guide
the name COGO = GCOGO.

EXPLANATION

COGO is a problem-oriented programming system that enables civil engineers without computer experience to solve coordinate geometry problems. The prime feature of the system is that engineers state problems in familiar terminology such as azimuth, deflection, and traverse adjustment.

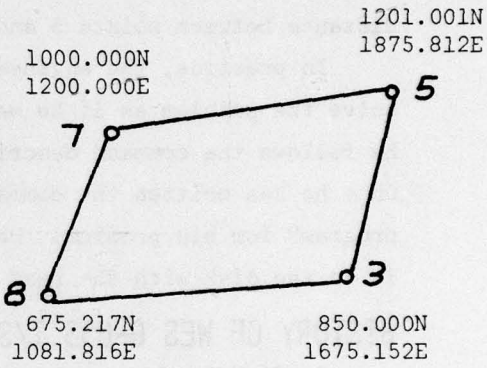
COGO can be applied to the computation problems involved in control surveys, highway design, right-of-way surveys, interchange design, bridge geometry, subdivision work, land surveying, and construction layout.

GENERAL CONCEPT

The COGO programming system is designed specifically for civil and structural engineering geometry problems. It may, however, be used in other application areas, in fact, there is almost no limit to the applicability of the system concept.

COGO is based on a vocabulary used by the engineer to state his problem. The statement of the problem is this familiar vocabulary and the input of these statements to the computer is all that is necessary to generate the solution to the problem. No programming, in the usual sense of the word, is necessary.

For example, an engineer interested in determining the area of the enclosed plat 7-5-3-8 states the problem as shown in Figure 1. The information in Figure 1 (with the exception of the diagram) is entered into the computer




```
STORE
4
7    1000.000  1200.000
3     850.000  1675.152
8     675.217  1081.816
5    1201.001  1875.812

AREA
5
7,5,3,8,7

AREA = 184463.3598 SQ. FT.,
       4.234696 ACRES

DISTANCE
5,8

FROM PT. 5 TO PT. 8  DIST = 952.3016
```

Figure 1

and the area is typed out automatically. One begins by giving the known information to the computer and then commanding it to perform specific functions on the known or previously calculated data. In this case the command AREA is used. It asks the computer to find the AREA of the enclosed polygon. Appearing right after the command is the result, so that the engineer can follow the sequence of calculation and keep a high degree of familiarity with the problem.

If the engineer wants the distance between points 5 and 8, he enters the command DISTANCE followed on the next line by 5 8. The distance between points 5 and 8 is then typed out by the computer.

In practice, the engineer uses a sketch and decides how to solve the problem as if he were analyzing it by hand. As a guide he follows the command descriptions shown later in this manual. Once he has written the commands on paper, he has a "computer program" for his problem. He then creates a data file and saves it on the disk with the read permission.

HISTORY OF WES G-635 T/S VERSION OF COGO

G-400 T/S source program was obtained from Polycom Systems Limited, a commercial time-sharing service company, Montreal, Canada, about 1969. This is a WES T/S program, converted from G-400 to G-600 by Dr. Howard Wilson and Dr. James Hill, University of Alabama, and William A. Price, Computer Analysis Branch (CAB), Waterways Experiment Station (WES).

BASIS OF SYSTEM

The COGO system is based on the repetitive use, by different commands in one program run or by several different runs, of a common data base. This data base contains coordinates of known and computed points and is known as the "Coordinate Table."

The engineer uses the COGO vocabulary to locate points on a traverse, subdivision, or along some alignment, etc. The points may be used in later calculations by other COGO commands and may be printed for

immediate use. The engineer gives each point an identification number whenever it is needed.

INPUT INSTRUCTIONS

After the engineering problem has been defined, the user must create a data file to instruct the computer in solving the problem. This data file consists of data sets with each set consisting of a command word and the numerical data needed to execute that command.

Command names may be used in either of two alternate forms: a long name of up to twelve characters or a short name not longer than four characters. Both forms must be started in the first character position of the command name field on the data line or card. These forms are shown in the table of contents.

The data may be entered in any one of three ways:

1. Time-Sharing, using a previously prepared data file. This file is free-field and must follow these rules:
   a. Each line starts with a line number followed by a single blank space. The first character of a command name must be in the first space following the single blank space: 1234 COMMAND.
   b. A command line may also contain any remarks to identify the reason for using the command, provided that the remarks start in at least the 15th space after the line number.
   c. Numeric data follows the usual FORTRAN rules: Items are separated by a comma or at least one blank space. The first value on a line must be after at least one blank space after the line number. Whole numbers need not be shown with a decimal point.
   d. It is important to give the exact number of data items called for on each line. In most instances, the numerical data needed following a particular command can be contained on a single line. In cases where

a group of data items is too long to go on one line, the number of values allowed on each line is specified according to the particular command used.

e. To avoid repetition, it is often desirable to reuse the last command entered. This is accomplished as follows. Assume that a line of numerical values is entered when a command line is expected. Then the program uses the last command given and employs the numeric data as the response which would logically follow after that command. For example, printing all points with indices between 12 and 25 would be accomplished by issuing the command DUMP followed on the next line by 12, 25. Now assume that three lines containing DUMP followed by 12, 25 followed by 30, 34 were entered. Points 12 through 25 would be printed. Then the numeric line 30, 34 would cause reuse of the DUMP command thereby causing points 30 through 34 to be printed.

2. Interactive Time-sharing, entering the commands and their numeric data as requested by questions typed out at the terminal. The arrangement of data on each line is identical to the input if from a data file, except that line numbers and the blank space after the line number are omitted. Lines of data should be entered from the terminal keyboard, following the equal sign typed by the program.
3. Batch Processing from either a remote site or at the computer center. Data format must follow these rules:
   a. Data will be on cards, in a free field format. Each card will have one line of data.
   b. No line numbers. Each command name must start in Column 1.
   c. The first command in the data deck must be preceeded by a card with -1 in the first two columns.
   d. If an END command in the data is followed by another data set in the same deck, then the END command card must be followed by a card with a 1 in column 1. If it is not

followed by more data, then it must be followed by a card with a 0 in column 1.

e. A card with zeros in columns 1 and 2, before any command card, will cause the printer to skip three extra lines. A blank card will cause it to skip one extra line.

f. Use this batch job control language card deck from a remote site:

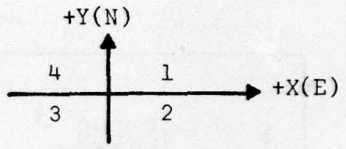
$ $ $ $ $ $ $ $ $ $ $	SNUMB USERID IDENT OPTION USE SELECT SELECT EXECUTE LIMITS DATA INCODE	XXXXX youruserid$password youruserid FORTRAN .GTLIT WESLIB/COGOCSTR WESLIB/COGOBSTR 20,28K yourkeypunchcode* 05
-1 first data END 0	command,	meaning "data in card deck" starting in column 1 last COGO command meaning "no more" problems
$	ENDJOB	

* The INCODE card is needed only if the COMMENT or REM command is used and if its data uses the special characters =, (,), or &.

DATA NOTATION

1. Bearings are entered by the quadrant method. A bearing is always stated in the format: quadrant number, degrees (integer), minutes (integer), and seconds (integer or decimal). Some commands, however, print the quadrant as letter pairs NW, SE, etc. or identify the quadrant separately.

2. Angles and azimuths. Angles and azimuths are entered as degrees, minutes, and seconds. For example, 75 0 5.0 is the code for 75° 0' 5.0". (However, degree of curvature is given in decimal degrees.)



Quadrants

Note that at least one blank column or comma separates degrees from minutes, and minutes from seconds. Degrees and minutes should be entered as integer quantities; the seconds may contain a decimal point and can contain decimal digits as well. Only the degrees portion carries a sign. Upon entry to the program, the user elects to measure azimuths as either (1) clockwise from south or (2) clockwise from north. The chosen option remains fixed during the subsequent problem analysis.

3. Zeroes and small negative angles. Zeroes must be included in the data. For example, an angle of zero degrees, zero minutes, and zero seconds, must be entered as 0 0 0.0. With counterclockwise angles (negative angles) less than one degree (for example, -0 12 27.0), use the 360 complement of the angle as the clockwise angle, since minus zero is not distinguishable from plus zero. In the example given, -0 12 27.0 must be entered as 359 47 33.0.

4. Coordinate system. COGO uses Y (North), X (East) coordinate system. Therefore coordinates must be entered in this order. The output is also given in this order.

5. Integers. Integer numbers must not contain a decimal point and may lie within the range of -999 and 999 except where otherwise restricted. The example shown with each command will indicate which parameters are integers and which are decimal numbers.

6. Point numbers. Point numbers must follow rule No. 5 for integers. They need not be in consecutive order, but missing point numbers being used must be included in the block of numbers specified in the preceding CLEAR command. Do not use zero as a point number and use only positive numbers.

STARTING A COGO RUN

1. Time-sharing

a. The following sequence of commands will start COGO time-sharing execution of the graphics version of COGO. User responses are underlined:

```
SYSTEM ?FORT N
READY
*RUN WESLIB/GCOGO,R
```

Execution starts with the lines

```
PROGRAM COGO --  USAE WATERWAYS EXPERIMENT STATION -- 08/27/79  -- 15.160

COORDINATE GEOMETRY ANALYSIS PROGRAM  733-F3-R0-  001A
REVISED  AUG 1979,  CORE SIZE = 30820 WORDS DECIMAL
```

where the "08/27/79" is the date of the run and "15.160" is the time (decimal hours on 24-hour clock) that the line was printed.

This is followed by the question

```
DATA INPUT FORM --
  ENTER 0 IF IN A TIMESHARING DATA FILE
    OR 1 IF RESPONSE TO QUESTIONS FROM THE TERMINAL
    (IT MAY ALSO BE RUN IN BATCH, WITH A CARD DECK)
=  (enter the appropriate answer after the = sign)
```

The following request to select azimuth reference is printed next:

```
ANGLE DATA CONVENTIONS --
  QUADRANTS  1 = NE  2 = SE  3 = SW  4 = NW
  SIGN = +  TO RIGHT, - TO LEFT
  SELECT AN OPTION DEFINING AZIMUTHS (1 = POSITIVE CLOCKWISE)
  FROM SOUTH, 2 = POSITIVE CLOCKWISE FROM NORTH)
=  (select the appropriate option)
```

If the question about data input form had been answered with a "1" for interactive input, the next message would be:

```
******************
*
* YOUR NEXT-TO-LAST COMMAND SHOULD BE "SAVE",
* SO YOU CAN RESTART LATER TO ADD OR CHANGE POINTS
*
*
******************
```

The final item in the initialization sequence, seen only if the question about data input form had been answered with a "0" for time-sharing data file input, is the following question and its answer:

ENTER THE FILE DESCRIPTION OF YOUR DATA FILE

= (enter the name and subcatalog name, if any, and passwords needed, in one of the following forms):

FILENAME

FILENAME$PASSWORD

CATALOG/FILENAME

CATALOG$PASSWORD/FILENAME

CATALOG$PASSWORD/FILENAME$PASSWORD

(DO NOT INCLUDE USER ID)

(47 CHARACTERS MAX)

The main program then starts to work, beginning with the first request for a command:

INPUT COMMAND WORD

=

b. If you are not sure about how to spell the command for a particular function, the program will consider a command of ? to be a request to print the command name table from the information file INFCOGO. When the table is finished, the program will ask for the next command entry. While this table is being printed, a single stroke of the BREAK or INTERRUPT key will stop the printing and go directly to the next command entry.

c. The last command must be END (or STOP). The program will then print out the question

DO YOU HAVE MORE DATA TO RUN?

(0 NO, 1 YES)

= (enter the appropriate answer)

An answer of "1" will return the program to its starting point initialization. If the program has been reading data from a file, the answer to the question must be in the data file line following the data line containing the word END.

If this last data file line contains an answer value of "1", then control is returned to the keyboard before the initialization sequence starts for the next set of data. A value of "0" stops the run.

2. Batch Processing

a. The job control cards are shown on page

b. The first data card must be preceeded by a card with its first two columns containing the value "-1" and the third column must be blank. The first data card must be a command name. Do not use line numbers; all command names must start in column 1.

c. The last data cards must be:

```
END
0
```

d. The SAVE and RESTORE commands for batch processing omit a filename line and place the first and last point numbers on the card immediately following the command name card. The complete batch processing RESTORE command data set will be as follows:

```
RESTORE
2  4
2  123.45  67.81
3  423.57  24.56
4  574.87  22.38
```

The SAVE command will punch a data deck that can be read by STORE or RESTORE commands in later runs.

GRAPHICS OPTIONS

When a Tektronix display terminal is employed, approximate graphical output for most of the COGO commands can be obtained. The command SHOW, followed by a numerical option equal to Ø, 1, or 2 controls the level of graphics output. Upon initiation of program execution, the SHOW option

is automatically set equal to Ø, corresponding to no graphics output. The option of 1 gives plotted results without numerical values, whereas an option of 2 gives both numerical and graphical results. The SHOW command can be given with different options at different stages in an analysis to regulate the amount of output. To employ the graphics features efficiently the user should know how the program functions when the screen becomes full. A line count is maintained to prevent print folding and overwriting of results already written. When the left bottom of the screen is reached, a program interrupt is generated requiring the user to (1) erase the screen and home to the upper left corner or (2) make a hardcopy before erasing and homing. The number of lines which may be written without filling the screen is based on the assumption that the smallest character size is being used. The terminal can be set to print small characters by switching to LOCAL and pressing ESC followed by a semicolon. When the terminal is returned to LINE, it will subsequently print only small characters.

Additional commands useful for controlling output are HDCOPY which copies the screen and ERASE which clears the screen and homes to the upper left corner. To avoid losing important results when the screen bottom is reached, it is advisable to use ERASE whenever values on the screen are no longer needed.

Using graphics involves two types of commands: those which have meaning only in a graphics context, and those which present a graphical interpretation of commands which are meaningful without graphics. Graphics output is plotted relevant to points lying within a window selected by giving the WINDOW command followed by the minimum Y,X and the maximum Y,X to be visible. When graphics results are plotted, only results inside the window appear on the screen. Giving the coordinate value of Ø,Ø,Ø,Ø after the WINDOW command, causes the existing point table to be scanned so that the range on all points is selected.

The following summary of commands refers to those which have a meaning exclusively in the context of graphics. Some of these commands such as ERASE require no additional data, whereas, a command like WINDOW is followed on the next line by 4 numbers defining the window limits.

Data: WINDOW or [WNDO]

YMIN,XMIN,YMAX,XMAX

This command sets the graphics window so that points lying within the specified range will be shown in subsequent graphics output. This command should always be given before any other graphics commands are employed. Otherwise, unpredictable results may occur due to incorrect scaling. When the values of YMIN,XMIN,YMAX,XMAX are taken as Ø,Ø,Ø,Ø then the complete coordinate table is scanned and the graphics window is set to encompass all existing points.

Data: SHOW

IPFLAG

This command controls the level of graphics output depending on whether IPFLAG is input as Ø, 1 or 2. IPFLAG = Ø gives no graphics output and display of numerical results is the same as earlier non-graphic versions of the program. IPFLAG = 1 gives graphic results without printing numerical values. IPFLAG = 2 gives both numerical results and plotted output.

Data: ERASE

Erase the screen and return the cursor to the upper left corner of the screen.

Data: HDCOPY

Make a hardcopy of the current screen image. Do not erase the screen.

Data: HOME

Move the graphic cursor to the upper left corner of the screen without erasing the screen. This command can be used when the screen is full but the user wishes to issue further commands without entering ERASE. However, after HOME is given the next command will overwrite any information present in the upper left corner of the screen.

Data: SKIP

NLINES

The line pointer is advanced NLINES from the current position.

Data: PAUSE

IREAD

The PAUSE command can be used to switch the reading of data input between a datafile and the terminal. The parameter IREAD is always read from the terminal. Taking IREAD equal to Ø or 1, respectively, causes subsequent input to be read from a datafile or from the terminal. For example, whenever the user wishes to switch from datafile input to terminal input, a PAUSE command can be contained in the datafile. When PAUSE is encountered then a value of IREAD is solicited interactively. If IREAD is entered as 1, then input continues to be solicited from the terminal. To revert to reading from the datafile, the program user can enter PAUSE followed by a zero value of IREAD. Then data and instructions will be read from the datafile starting at the current position in the file.

Data: SHOW/N-PTS or [SHNP]

NPTS [number of points]

N1,N2, . . . ,N1Ø [not more than ten points per line]

The indices for a total of NPTS points to be shown are entered using one or more data lines. Not more than ten point indices are entered on a line. When the second data parameter NPTS is entered as zero, then all points in the data table are shown.

Data: SHOW/R-PTS or [SHRP]

N1,N2

All points with indices between and including N1 and N2 are plotted.

Data: LINE/N-PTS or [LINP]

NPTS [number of points]

N1,N2, . . . ,N1Ø [not more than ten points per line]

The indices for a total of NPTS points to be shown are entered using one or more data lines. Not more than ten point indices are entered on a line. The plotted points are connected by straight lines. When NPTS is given as zero then all points in the data table are shown.

Data: *LINE/R-PTS or [LIRP]*

N1,N2

All points with indices between and including N1 and N2 are plotted and are appropriately connected by straight lines.

DESCRIPTION OF COMMANDS

The following group of commands is used to define and solve the geometry associated with an alignment, including simple curves, tangents, offsets, and stationing along the line. All curves are circular, and stations are expressed in decimal feet (station 80 + 23 is entered as 8023).

An ALIGNMENT or DEFCV command must precede any of the following commands:

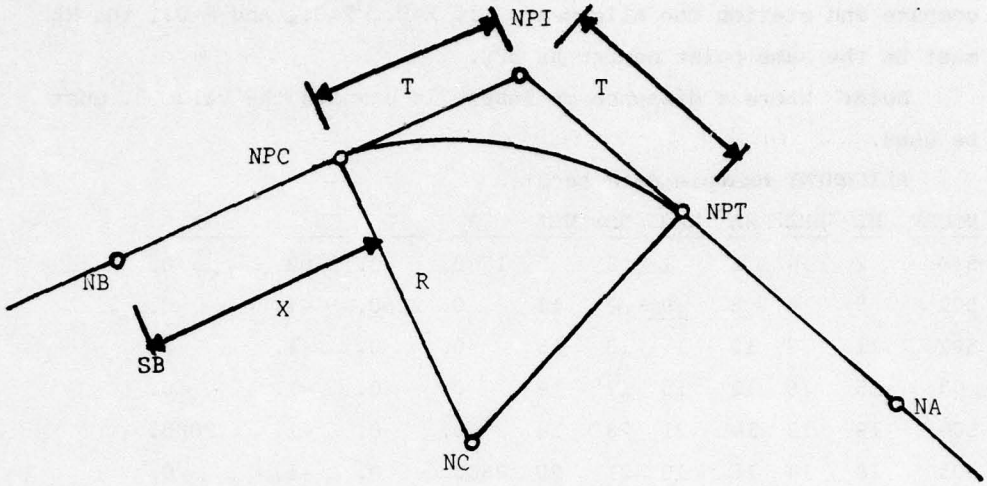
Full Name	Short Name
COPOA	COPO
COOFF	COOF
STAFC	STAF
OFSAL	OFSA
DVLIN	DVLI

The ALIGNMENT and DEFCV commands establish a curve by giving it a number and storing its parameters. As many as 25 curves may be stored. However, only one curve at a time can be active. All of the above commands apply to the most recently created curve or to one retrieved by the GETCRV command. A curve no longer needed is removed by the DLTCRV command.

If all the data for a curve is known, then the DEFCV command should be used. However, if the curve has some unknown quantities, then the ALIGNMENT command is appropriate.

Data: ALIGNMENT or [ALIG]

NCURV NB NPI NA NPC NC NPT R T SB X



Compute the curve, given the following:

NCURV Identification number of curve (0-999).

NB Any known point on back tangent.

NPI Known point of intersection of the tangents, PI.

NA Any known point on ahead tangent.

NPC Number assigned to the beginning of the curve that is, the point of curvature, PC.

NC Number assigned to center of curve.

NPT Number assigned to the end of the curve, that is, the point of tangency, PT.

R Radius of curve (if unknown, 0.).

T Tangent length of curve (if unknown, 0.).

SB Station at NB. If entered as -1., NB is taken to be, and must be entered as, the PT of previous curve, and SB is taken as the station of the previous PT (that is, X=0). This allows stationing to be automatically carried forward (see the example on next page).

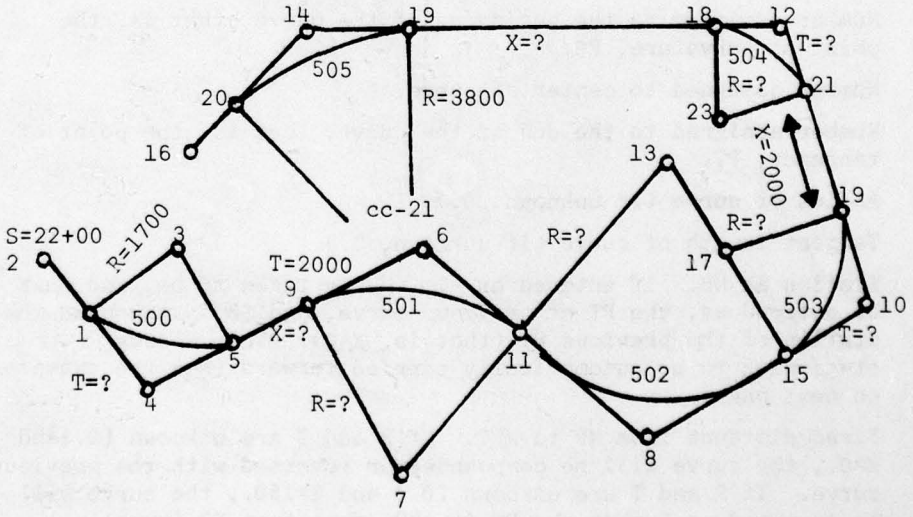
X Fixed distance from NB to NPC. If R and T are unknown (0.)and X=0., the curve will be compounded or reversed with the previous curve. If R and T are unknown (0.) and X=150., the curve will be computed such that the PC is 150 feet from NB (usually but not necessarily the NPT of the previous curve).

All PI's must be located before using ALIGNMENT command to compute and station the alignment. If X=0., T=0., and R=0., the NB must be the same point number as NPC.

Note: Where a distance or length is unknown the value 0. must be used.

ALIGNMENT example data sets:

NCURV	NB	NPI	NA	NPC	NC	NPT	R	T	SB	X
500	2	4	6	1	3	5	1700.	0.	2200.	0.
501	5	6	8	9	7	11	0.	2000.	-1.	0.
502	11	8	10	11	13	15	0.	0.	-1.	0.
503	15	10	12	15	17	19	0.	0.	-1.	0.
504	19	12	14	21	23	18	0.	0.	-1.	2000.
505	18	14	16	19	21	20	3800.	0.	-1.	0.
599	1	4	6	2	3	5	0.	0.	0.	100.



500 is an example of an independent curve without any relationship to a previous curve.

501 is an example of an alignment curve defined by tangent length and with stationing carried forward from previous curve.

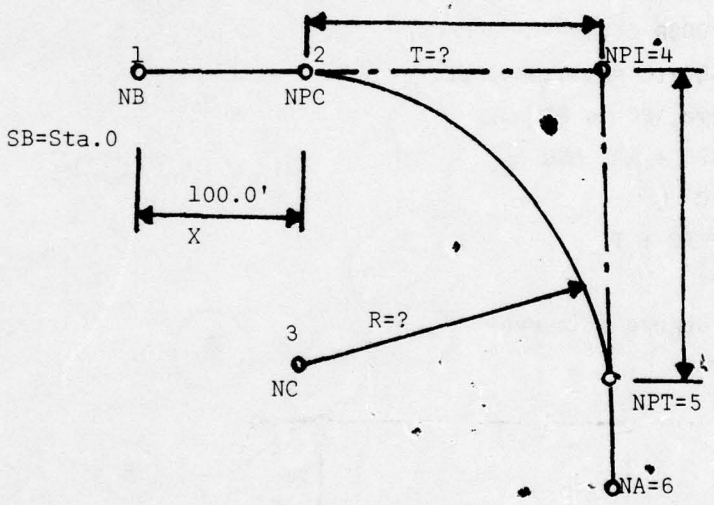
502 is an example of a reverse curve.

503 is an example of a compound curve.

504 is an example of an alignment curve fixed by a tangent section of specified length.

505 is an example of an alignment curve defined by radius and stationing carried forward.

Sample problem to be run:



known points: 1,4,6
T=?
points to be computed: 2,3,5
T,R

ALIGNMENT command output from sample :

```
ALIGNMENT

     599 (1.)  R=    400.0000  T=    400.0000    ANGLE   90    0    0.0
LONG CORD=    565.6854    MID ORD=    117.1573    EXT DIST=    165.6854
X=  100.00000  SPC=    100.0000  ARC LNG=    628.3185  SPT=    728.3185
POC    2    Y=    600.0000    X=    300.0000    SPI=    500.0000
POT    5    Y=    200.0000    X=    700.0000   DEG=    14    19    26.2
CTR    3    Y=    200.0000    X=    300.0000
POI    4    Y=    600.0000    X=    700.0000
```

Explanation of output from sample problem:

Curve identification number = 599.

(1.) indicates a curve to the right. (-1.) would have indicated a curve to the left.

Radius required to fit stored locations of NB, NPI, and NA; and the input value of X=400.0000ft.

Central angle = 90°

Long cord = straight line from PC to PT = 565.6854 ft.

Mid ordinate from long cord to curve = 117.1573 ft.

External distance from curve to PI = 165.6854 ft.

X = input value of 100.0000 ft.

SPC = Station of PC = SB + X = 100.0000 ft.

ARC LNG = Length of curve, PC to PT.

SPT = Station of PT = SPC + ARC LNG

POC = Coordinates of NPC

SPI = Station of NPI = SPC + T

POT = Coordinates of NPT

DEG = (arc definition) degree of curve

CTR = coordinates of NC

POI = Coordinates of NPI

Data: DEFCV or [DEFC]

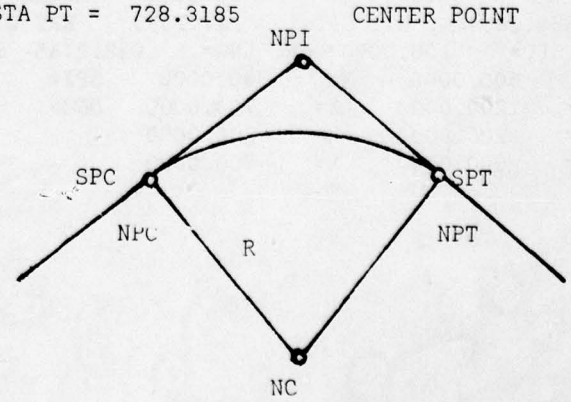
NCURV NPC SPC NPI NPT SPT NC SIGN

Example output (summary of input):

CURVE NO. 599 SIGN = 1 (DEFINED)

PC POINT 2 STA PC = 100.0000 PI POINT 4

PT POINT 5 STA PT = 728.3185 CENTER POINT 3



NCURV	Curve number (0-999).
NPC	Number assigned to the beginning of the curve, that is, the point of curvature, PC.
SPC	Station of the PC.
NPI	Known point at the point of intersection of the tangents, PI.
NPT	Known point at the end of the curve, that is, the point of tangency, PT.
SPT	Station of the PT.
NC	Number of the known center of the curve.
SIGN	1.0 for clockwise curve (from PC to PI), -1.0 for counterclockwise curve (from PC to PI).

Data: GETCRV

NUMBR

This command retrieves from storage a particular curve having an index value of NUMBR. The parameters of the curve are obtained and appropriate calls to ALIGN or DEFCV are executed depending on how the curve was defined originally. Only one curve at a time can be retrieved.

Data: LSTCRV

NCRVS

NUMBR (1),···, NUMBR (NCRVS)

The various parameters on at total of NCRVS curves are listed. These curves have indices NUMBR (1) through NUMBR (NCRVS). If the value of NCRVS is entered as zero then all curves are listed.

Data: DLTCRV

NCRVS

NUMBR (1),···, NUMBR (NCRVS)

A total of NCRVS are deleted from the curve table. If NCRVS is zero, then all curves are deleted. Otherwise, the indices of curves to be deleted are given on subsequent data lines as NUMBR (1),···, with up to ten curve indeces on a line.

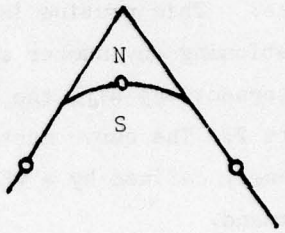
Each of the four following routines automatically selects the back tangent, curve section, or forward tangent, whichever is appropriate.

Data: COPOA or [COPO]
N S

Compute the coordinates of point N on the alignment at station S.

Output: Coordinates of N.

Note: The curve must have been previously defined by: DEFCV or ALIGNMENT command.



Example output:

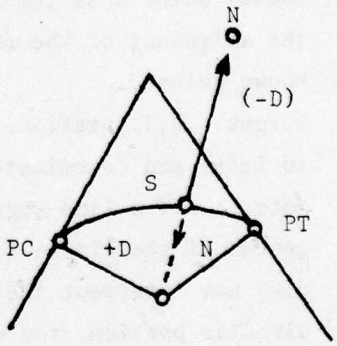
```
COPOA
PT.      8      Y=      537.5650      X=      398.9616
```

Data: COOFF or [COOF]
N S D

Compute the coordinates of point N at station S and an offset distance D (minus if to the left).

Output: Coordinates of N.

Note: The curve must have been previously defined by a DEFCV or ALIGNMENT command.



Example output:

```
COOFF
PT.      9      Y=      490.6737      X=      374.2212
```

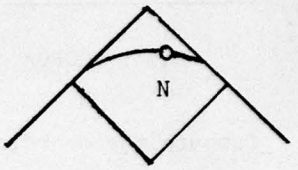
Data: STAFC or [STAF]
N

Compute the station of known point N on the alignment.

Output: Point number and station of N.

Note: This routine is useful in stationing any number of points located by intersections with the center line.

Note 2: The curve must have been previously defined by a DEFCV or ALIGNMENT command.



Example output: STAFC

10 728.3185

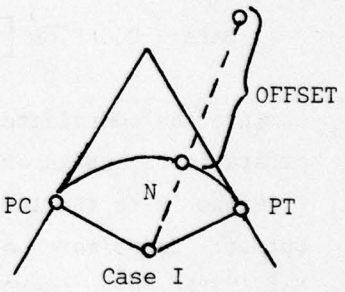
Data: OFSAL or [OFSA]
N J

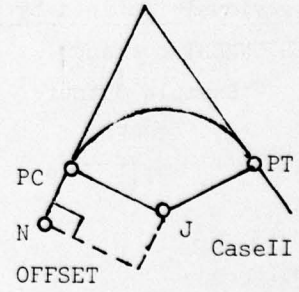
Locate point N as the intersection with the alignment of the radial offset from known point J.

Output: N,J. station, offset (minus if to left) and coordinates of N.

Note 1: If a line segment drawn from the center of the circle to J and beyond it does not intersect the alignment in the circular portion, the offset will always be to the line of the back tangent (see case II).

Note 2: The curve must have been previously defined by a DEFCV or ALIGNMENT command.





Example output:

OFSAL

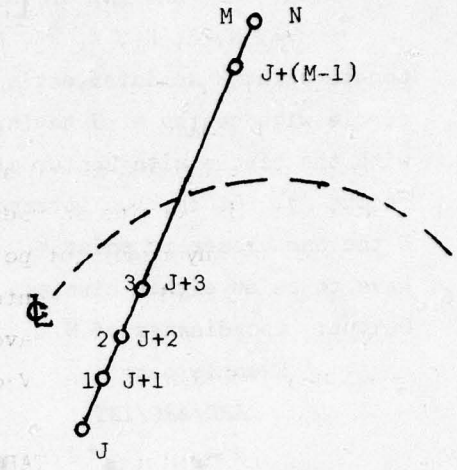
PT. 12 PT 11 STA. 414.16 OFFSET -165.6854

Data: DVLIN or [DVLI]
J N M

Divide the line between defined points J and N into M equal parts. The intermediate points are automatically assigned point numbers J+1, J+2, J+...+(M-1).

Output: Coordinates of each intermediate point. Center line station and offset for each point to stores curve.

Note: Point numbers are assigned successively from J. Because of this, care should be exercised to avoid destroying known points by careful selection of point J.



Example output:

DVLIN

incremental distance

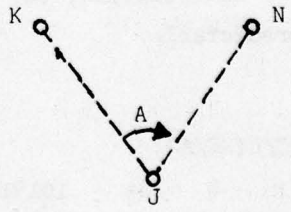
141.4214

PT. 22 Y= 200.0000 X= 100.0000
STATION -100.0000 OFFSET= 400.0000
PT. 23 Y= 300.0000 X= 200.0000
STATION 0.0000 OFFSET= 300.0000
PT. 24 Y= 400.0000 X= 300.0000
STATION 100.0000 OFFSET= 200.0000

Data: ANGLE or [ANG]
K, J, N

Compute the clockwise angle at J from K to N.

Output: Angle A at J from K to N.



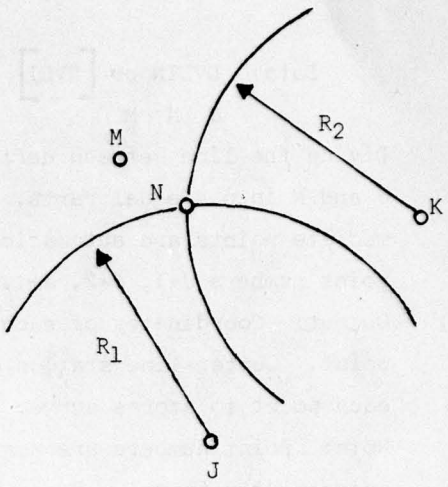
Example output:

ANGLE
CLOCKWISE ANGLE FROM K 6,AROUND J 8, TO N 1, = 270 0 0.

Data: ARC/ARC/INT or [A/AI]

N, J, R_1, K, R_2, M

Locate point N as intersection of the circle with center at J having radius R_1 with the circle with center at K having radius R_2. Of the two intersections, save as N the one closer to point M. (M does not have to be on either circle.)



Output: Coordinates of N.

Example output

ARC/ARC/INT

N	J	R_1	K	R_2	M
21	11	141.4200	12	141.4200	14

PT. 21 (N) Y= 99.9981 (Yn) X= 400.0000 (Xn)

Data: ARC/LINE/AZ or [A/LA]

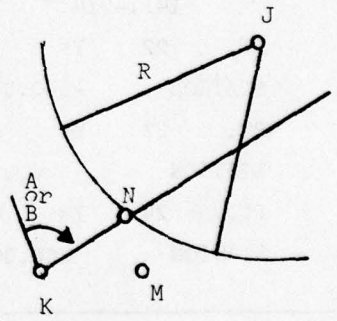
N, J, R, K, A, M

Data: ARC/LINE/BR or [A/LB]

N, J, R, K, B, M

Same as above, except that the line is defined by point K and azimuth A or bearing B.

If M is equidistant from the two points of intersection, the results are unpredictable.



Angle Data:

A: DEG,MIN,SEC
B: QUAD,DEG,MIN,SEC

(DEG & MIN are integers)

ARC/LINE/AZ

PT. 6 Y= 10151.0092 X= 10041.7603

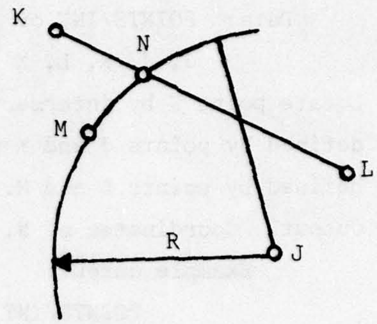
ARC/LINE/BR

PT. 22 Y= 200.0000 X= 100.0000

Data: ARC/LINE/PT or [A/LP]

N, J, R, K, L, M

Locate point N as the intersection of the circle with center at J having a radius R with the line defined by points K and L. Of the two intersections save as N the one closer to point M. (M does not have to be on the circle.)



Output: Coordinates of N.

Example output:

ARC/LINE/PT

PT. 9 Y= 0. X= 400.0000

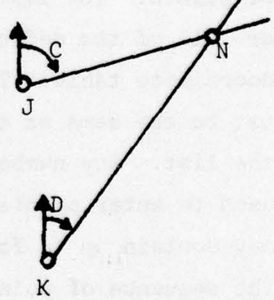
Data: AZIMUTH/INT or [A/I]

N, J, C, K, D

Data: BEARING/INT or [B/I]

N J C K D

Locate point N by intersecting the line defined by point J having *azimuth* (bearing) C with the line defined by point K having azimuth (bearing) D.



Output: Coordinates of N.

Example output:

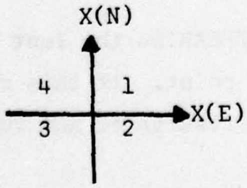
AZIMUTH/INT

PT. 4 Y= 10217.1560 X= 10060.0527

BEARING/INT

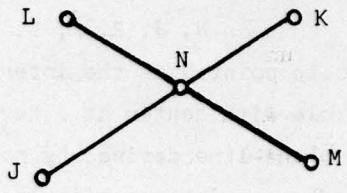
N → 7 J ↙ 8 quad 3 45° 0' 0." (C) K ↘ 6 quad 4 45° 0.' 0." (D)

PT. 7 Y= 200.0000 X= -0.0000



Bearing Quadrants

Data: POINTS/INT or [P/I]
J, J, K, L, M



Locate point N by intersecting the line defined by points J and K with the line defined by points L and M.

Output: Coordinates of N.

Example output:

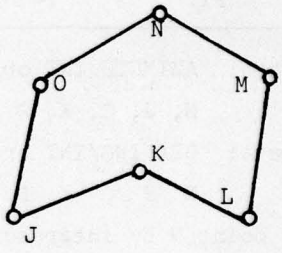
POINTS/INT
N 14 J 6 K 1 L 8 M 5

PT. 14 Y= 200.0000 X= 400.0000
N

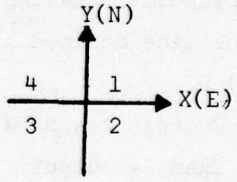
Data: AREA

Data: AREA/AZIMUTH

Data: AREA/BEARING



Compute the area enclosed by the list of points. The list may include any or all of the defined points in the coordinate table. The last point in the list ust be the same as the first point in the list. Any number of lines may be used to enter points and each line may contain up to fourteen points. The sequence of points may be in either direction.



Bearing Quadrants

Output: Area enclosed by list of points is square feet and acres. AREA/AZIMUTH also gives table of distances and azimuths of each side of the polygon. AREA/BEARING also gives table of distances and bearings of each side of the polygon. The cumulative area counter for later SEGMENT/PL and SEGMENT/MI commands is set at the answer of the AREA commands.

Note: In the commands AREA, AREA/AZIMUTH,and AREA/BEARING the last point entered must always be the same as the first point. If this rule is not followed there is no error message, and the results of any further

area commands or any succeeding adjust commands will be unpredictable. At this point the user should stop and start over at the beginning of the run.

Samples:

AREA data: AREA or [AR]

```
5
6  4  5  7  6
```

Example output:

AREA

6 4 5 7 6
11069.40540 SQFT. 0.25412 ACRES

AREA/AZIMUTH data: AREA/AZIMUTH or [AR/A]

```
5
15  1  8  20  15
```

Example output:

AREA/AZIMUTH

from	to	distance	azimuth 0	1	11
15	1	200.0000	0	0	0.0
1	8	400.0000	270	0	0.0
8	20	200.0000	180	0	0.0
20	15	400.0000	90	0	0.0

79999.9970 SQFT. 1.8365 ACRES

AREA/BEARING data: AREA/BEARING or [AR/B]

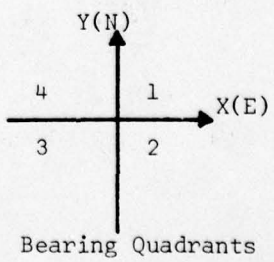
```
5
6  4  5  7  6
```

Example output:

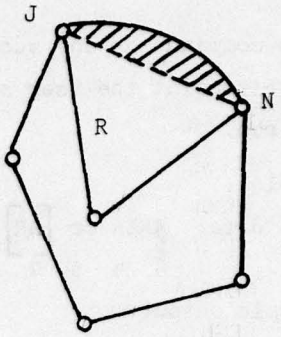
AREA/BEARING

from	to	distance	quad	0	1	11
6	4	68.6296	1	15	27	30.0
4	5	185.7120	2	90	0	0.0
5	7	47.1536	2	4	59	40.0
7	6	208.9908	3	84	44	11.2

11069.4158 SQFT. 0.2541 ACRES



Bearing Quadrants



Data: SEGMENT

J, N, R

Data: SEGMENT/PL

J, N, R

Data: SEGMENT/MI

J, N, R

For the segment of a circle defined by points J and N on the circumference and with radius R, compute the area.

Output: The chord length, arc length, segment area in square feet and acres. For SEGMENT/PL or MI the area is added to or subtracted from the cumulative net area resulting from the last AREA type command and the previous SEGMENT/PL or MI commands. This allows any number of segments to be added to or subtracted from a polygon for parcels bounded by curves. After each SEGMENT/PL or MI, the net area up to that point is recorded.

Note: Only the first 7-1/2 digits of the answers have any meaning.

Examples:

SEGMENT data: SEGMENT or [SEG]

11 21 100.0

Example output:

```
SEGMENT
                          J          N          R
THROUGH POINTS     11     AND     21     RADIUS = 100.00000
CHORD=        70.7114     ARD=      157.0777
CENTRAL ANGLE=     89     59     56.0
SEGMENT AREA=     2853.88573  SQFT.          0.06552  ACRES
```

SEGMENT/PL data: SEGMENT/PL or [S/P]
11 21 100.0

Example output:

```
SEGMENT/PL
THROUGH POINTS  11  AND   21  RADIUS = 100.00000
CHORD=       70.7114     ARC=       157.0777
CENTRAL ANGLE=    89      59      56.0
SEGMENT AREA=    2853.88573  SQFT.        0.06552 ACRES
NET AREA=     82853.88269  SQ. FT. =       1.90206 ACRES
```

SEGMENT/MI data: SEGMENT/MI or [S/M]
11 21 100.0

Example output:

```
SEGMENT/MI
THROUGH POINTS   11 AND    21    RADIUS = 100.00000
CHORD=         70.7114    ARC=     157.0777
CENTRAL ANGLE=     89    59     56.0
SEGMENT AREA=     -2853.88573 SQFT.        -0.06552 ACRES
NET AREA=        79999.99696 SQ. FT. =         1.83655 ACRES
```

Data: CLEAR or [CLR]
I, K

Clear the coordinate table from point I to point K. This command places a very large negative coordinate value in each entry in the table. This should always be the first command when starting a new job, and the entire table to be used should be cleared. To clear one point, use the point number twice.

Example output:

```
CLEAR
BETWEEN POINTS    1 AND  20
```

Data: COMMENT or [REM]
up to 80 alphanumeric characters of remarks

Output: printing of remarks

Single data line: -- (two dashes)

Output: skips 3 extra blank lines (usually precedes a comment)

Data: DISTANCE or [DIST]
J, N

Compute the distance between points J and N.

Output: Distance from J to N.

Example output:

```
DISTANCE
FROM PT.    7 TO PT.    8  DIST=    282.8427
```

Data: DUMP or [LIST]
I, K

Print the coordinates of all <u>defined</u> points from point I through point K. Points not defined will be omitted.

Example output:

DUMP

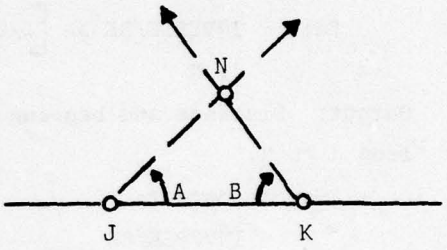
POINT	Y	X
1	400.0000	600.0000
2	200.0000	800.0000
3	100.0000	600.0000
4	100.0000	600.0000
5	0.	600.0000
6	0.	200.0000
7	200.0000	-0.0000
8	400.0000	200.0000
9	0.0000	400.0000
10	400.0000	0.

Data: END or [STOP]

This command should be the last statement in the last job so as to terminate the computer processing and signify that no more COGO commands follow. If a batch job, the END card must be followed by a blank card or a card with a zero in column 1.

Data: FORSECTION or [FORE]
N J A K B

Locate point N by intersecting a line defined by turning an angle A (minus if counterclockwise) from base line point J to N with a line defined by turning an angle B (+ as shown) from base line point K to N.



Example output:

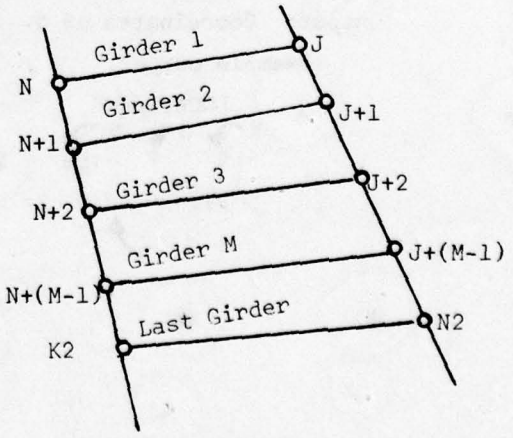
```
FORSECTION

    20    7  -45.0     0.     0.     6  45.0     0.     0.
       200.0000          200.0000          282.8427

PT.    20    Y=    200.0000    X=    200.0000
```

Data: GIRDL or [BEAM]
J N2 M N K2

The incremental distances N to N+1, N+1 to N+2, etc, need not be equal. All points must have been defined earlier in the run. The variable "M" is the number of spaces between girders, or one less than the number of girders.



Example output:

GIRDL

PT	TO PT	DISTANCE
1	2	1.4142
2	3	4.2426
3	4	2.8284
4	5	1.4142

Data: INVERSE/AZ or [I/A]

J, N

Output: Distance and azimuth of the line from J to N.

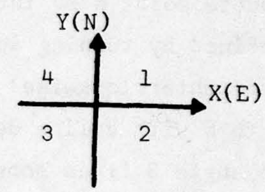
Example output:

```
INVERSE/AZ
FROM      7 TO    8, DIST =     282.8427
AZIMUTH =   45    0    0.0
```

Data: INVERSE/BE or [I/B]

J, N

Output: Distance and bearing of the line from J to N.



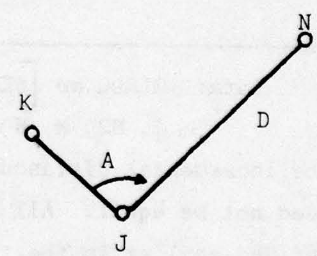
Bearing Quadrants

Example output:

```
INVERSE/BE
FROM         7 TO    8,  DIST=     282.8427
BEARING =      45    0    0.0    IN QUADRANT 1
```

Data: LOCATE/ANG or [L/AN]

K, J, N, D, A



Backsight on K, turn angle at J to locate N at a distance D and angle A (degrees, minutes, seconds). Angle may be clockwise or counterclockwise (minus A).

Output: Coordinates of N.

Example output:

```
LOCATE/ANG
K    J    N      D                    A
6    7    10     200.0000      0     1    11
                              -135    0    0.
PT.       10   Y=    400.0000    X=     0.
          N
```

Data: LOCATE/ANG2 or [LAN2]

K, N, J, D, A, NERFAR

Backsight on K, locate a new point N at distance D from J such that the angle turned at N equals A (degrees, minutes, seconds). When D exceeds the distance from J to K then two solutions exist. Choosing parameter NERFAR equal to 1 or 2 gives, respectively, the solution point nearer or farther from K.

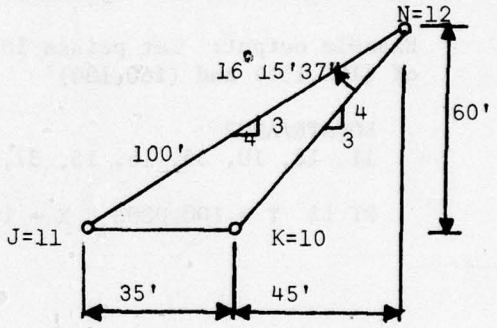
Output: Coordinates of N

Example output: Let points 10 and 11 have (Y,X) coordinates of (100,135) and (100,100)

LOCATE/ANG2

10, 12, 11, 100, 16, 15, 37, 1

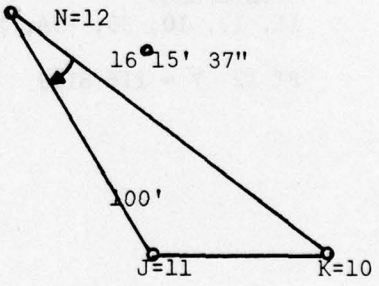
PT 12 Y = 160.0004 X = 179.9997



LOCATE/ANG2

10, 12, 11, 100, 16, 15, 37, 2

PT 12 Y = 193.6003 X = 64.8007



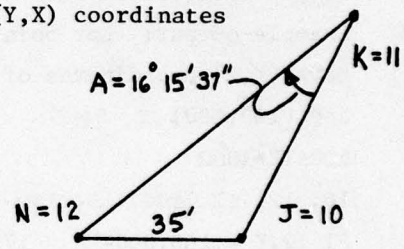
Data: LOCATE/ANG3 or [LAN3]

K, N, J, D, A, NERFAR

Backsight to point J from point K. Locate a new point N at a given distance D from J. Let the angle A turned at K from J to N have a specified value. Angle A (degrees, minutes, seconds) may be clockwise or counterclockwise (minus A). When D is smaller than sin(A) times the distance from J to K, then no solution exists. Otherwise, two solutions are possible. Taking parameter NERFAR equal to 1 or 2 gives, respectively, the point N which is nearer or farther from K.

Output: Coordinates of point N

Example output: Let points 10 and 11 have (Y,X) coordinates of (100,135) and (160,180)

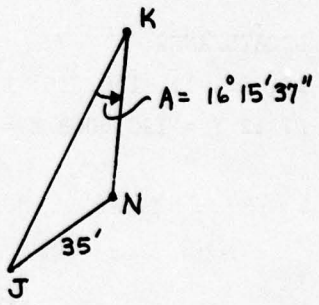



```
LOCATE/ANG3
11, 12, 10, 35, 16, 15, 37, 2

PT 12  Y = 100.0002   X = 100.0000
```

```
LOCATE/ANG3
11, 12, 10, 35, -16, 15, 37, 1

PT 12  Y = 118.8159   X = 164.5120
```



Data: LOCATE/AZI or [L/A]

J, N, D, A

From J locate N at a distance D and azimuth A (degrees, minutes, seconds). Angle may be clockwise or counterclockwise (minus A).

Output: Coordinates of N.

Example output:

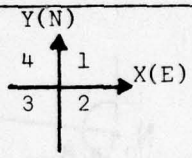
```
LOCATE/AZI
J    N    D          A
4    5    185.7120   90   0   0.

PT.   5   Y=   10217.1560   X=   10245.7647
```

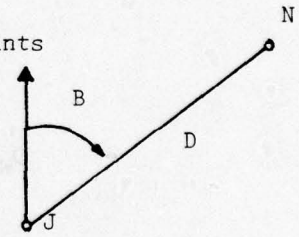
Data: LOCATE/BEA or [L/B]

J, N, D, B

From J locate N at a distance D and bearing B (degrees, minutes, seconds). The angle measured from north or south.



Bearing Quadrants



Output: Coordinates of N.

Example output:

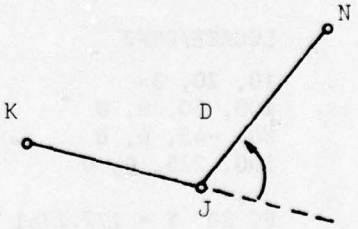
```
LOCATE/BEA
6    11    100.00000   1   90   0   0.

PT.   11   Y=   0.   X=   000.0000
```

Data: LOCATE/DEF or [L/D]

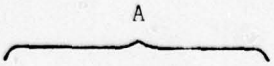
K, J, N, D, A

Backsight on K, turn deflection angle at J to locate N at a distance D and deflection angle A. (Degrees, minutes, seconds). Angle may be clockwise or counterclockwise (minus A).



Output: Coordinates of N.

Example output:




```
LOCATE/DEF
K    J    N          D
6    5    4          100.0000   -90   0   0.

PT.   4   Y=   100.0000   X=   600.0000
```

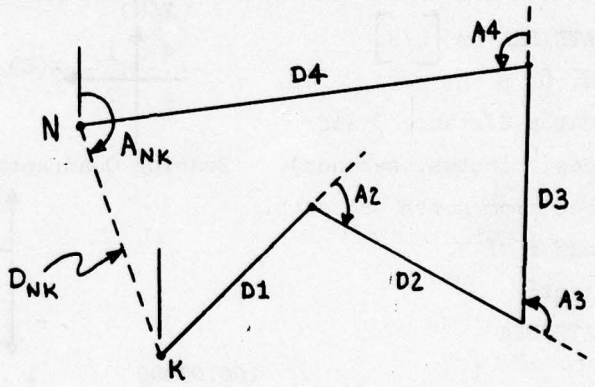
Data: LOCATE/DEF2 or [L/D2]

K, N, M

DIST, ANGLE
.
.
DIST, ANGLE } total of M lines

Starting from a known point K, determine a new point N by projecting forward M courses. Output includes the coordinates of N, along with the length and azimuth of the line from N to K. Each course is determined by a distance and an angle (degrees, minutes, seconds). The angle for the first course is the azimuth. Subsequent angles are deflection angles which can be positive or negative (counterclockwise).

Output: Coordinates of N and the length and azimuth of the line from N to K.

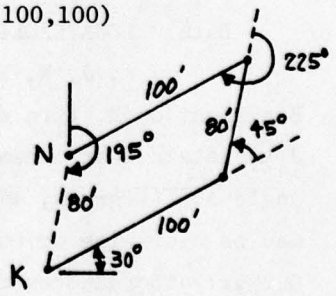


Example output: Let point K have coordinates of (100,100)

```
LOCATE/DEF2

10, 20, 3
100, 60, 0, 0
80, -45, 0, 0
100, 225, 0, 0

PT 20  Y = 177.2741      X = 120.7055
```

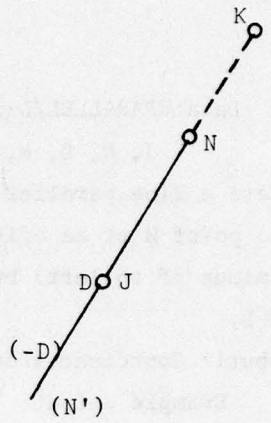


Data: LOCATE/LIN or [L/L]

J, K, N, D

From J, in the direction of K, locate N at a distance D from J (minus D would locate N).

Output: Coordinates of N.



Example output:

```
LOCATE/LIN
  J      K      N        D
  6     11      5    400.0000

PT.      5     Y=        0          X=        600.0000
```

Data: TANGENT/OFF or [T/O]

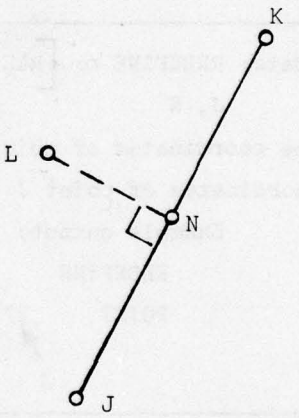
N L J K

Compute the point N at the intersection of the perpendicular offset from point L to the line J, K.

Output: Coordinates of point N

Distance J to N

Distance L to N

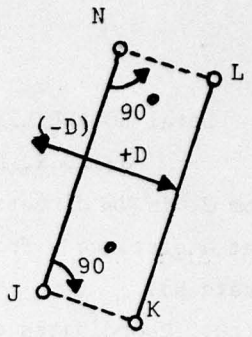


Example output:

```
TANGENT/OFF
PT.      15      Y=      200.0000      X=        600.0000
FROM PT.      4 TO PT.    15    DIST=       100.0000
FROM PT.      2 TO PT.    15    DIST=       200.0000
```

Data: PARALLEL/L or [P/L]

J, N, D, K, L



Locate a line parallel to the line from point J to point N at an offset distance D (minus if to left) by locating points K and L.

Output: Coordinates of K and L.

Example output:

```
PARALLEL/L
J→ 6      8←N-      D→ 300.0000      K→ 12      L→ 13
PT.   12   Y=        0.        X=      500.0000
PT.   13   Y=      400.0000    X=      500.0000
```

Data: REDEFINE or [RDEF]

J, N

Set the coordinates of point N equal to the coordinates of point J.

Example output:

```
REDEFINE
POINT     17    IS NEW LOCATION FOR POINT   9
          ↑J                                ↑N
```

Command RESTORE or [REST]

Timesharing data: RESTORE
AFILE
J K

Timesharing action: Attaches disc file named AFILE, searches through to find point J, reads coordinate data from J until it finds point K, reads point K. detaches AFILE. Point data read is stored in Coordinate Table.

Batch data: RESTORE
J K
J Y(J) X(J)
...
K Y(K) X(K)

Batch action: Searches following coordinate cards for point J, reads from J to K, stores the values read into Coordinate Table.

Note: Point K must be the last card before the next command.

Command SAVE or [SAVE]

Timesharing data: SAVE
AFILE
J K

Timesharing action: Attempts to attach the file (creates a new file if no file by that name exists); writes the portion of the Command Table between points J and K, inclusive, into file AFILE; detaches the file AFILE for later use in a RESTORE command.

Batch data: SAVE
J K

Punches cards containing the portion of the Coordinate Table between points J and K, inclusive. Card format is readable by STORE or RESTORE commands. Puts card with all + signs after last card in Table.

Data: STORE or [STO]
N
I Y(I) X(I)

N = number of point coordinate sets to be stored.

I = point number.

One such line of data for each point to be stored into Coordinate Table. The points need not be in any particular order. There must be N lines.

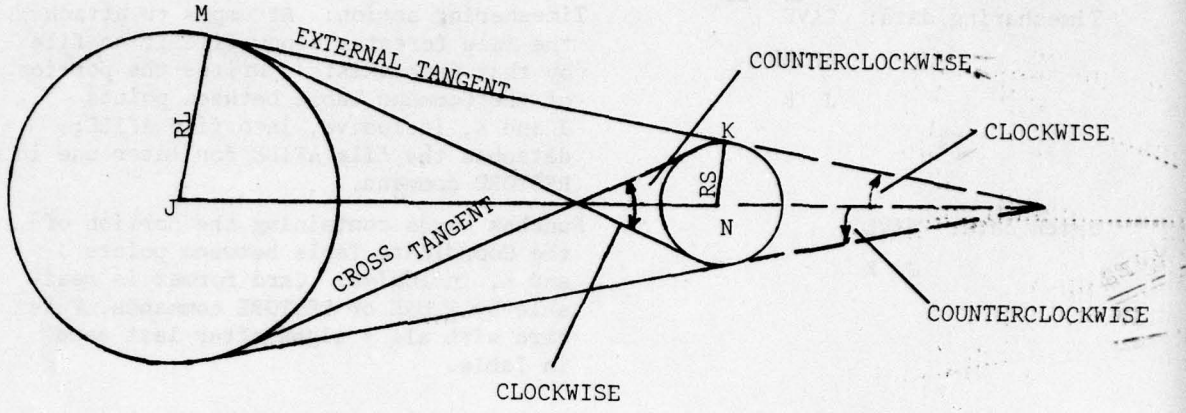
Note: In batch, this command will read cards punched by a SAVE command in a previous run.

Data: TANGENT or [TAN]
M, J, RL, K, N, RS, SIGN, CROSS

Locate M and K as the two end points connecting the tangent to the two circles with centers J and N and radii of RL and RS, RL and J being associated with the larger of the two circles.

SIGN is used for the selection of one of two possibilities in either case. SIGN is 1, when the angle formed by the extension of the line connecting the centers of the two circles and the extension of the desired tangent is clockwise. It is -1. when the angle is counterclockwise The sense of the angle is from the connecting line to the desired tangent. CROSS is designed as 1. for the selection of an exterior tangent and -1. for the selection of a cross tangent. The larger circle must be entered first.

Output: Coordinates of M and K. Distance and azimuth of the tangent from M to K.



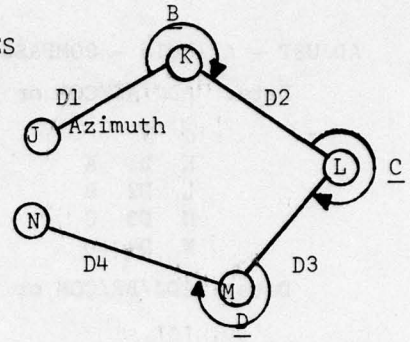
Example output:

```
        TANGENT
        PT.     18      Y=      296.8246        X=      425.0000
        PT.     19      Y=      248.4123        X=      612.5000
           18       19      193.6492        1    49     24.5
                                            0    1      11
```

distance M-K

azimuth of line from M to K

TRAVERSE ADJUSTMENT ROUTINES



ADJUST - ANGLE - COMPASS method:

Data: ADJUST/ANG or [A/A]

J N
J K DI A
L D2 B
M D3 C
N D4 D

Adjust-angle-compass method. Adjust the angle traverse which originates on point J and closes on point N. J and N may be the same point. For the first course, the point back (J), the point ahead, the distance, and the clockwise angle at the point back is given. The coordinates of the adjusted survey points are computed and stored.

Example output:

```
ADJUST/ANG
 J→3   N→3      Y coord's        X coord's
 K→4        100.0000         300.0000
 L→1        300.0000         300.0000      } coordinates calculated from
 M→2        300.0000         100.0000      } input data (not stored)
 N→3        100.0000         100.0000
Y ERROR=      -0.0000  X  ERROR=     0.0000  TOT. ERR=     0.0000
AZ OF C.L. = 135    0    0.0   PERIM=    800.000  RATIO=1/0.27896 E 08
     4        100.0000         300.0000      } coordinates of adjusted points (stored)
     1        300.0000         300.0000      }
     2        300.0000         100.0000      }
PT TO PT    DISTANCE        ANGLE
     3    4     200.0000     90    0    0.0   }
     4    1     200.0000      0    0    0.0   } adjusted traverse
     1    2     200.0000    270    0    0.0   }
     2    3     200.0000    180    0    0.0   }
     3    3       0.         90    0    0.0   }
                             0    1   11
pt. back    pt. ahead
```

lft of error for every 13,635,000 ft. of perimeter

ADJUST - AZIMUTH - COMPASS method and ADJUST - BEARING - COMPASS method:

Data: ADJ/AZ/COM or [A/AC]

```
J  N
K  D1  A
L  D2  B
M  D3  C
N  D4  D
```

Data: ADJ/BR/COM or [A/BC]

```
...
...
```

Adjust the azimuth (or bearing) traverse which originates on point J and closes on point N. J and N may be the same point. For each course, the point ahead, the distance, and the azimuth (or bearing) are given. The coordinates of the adjusted survey points are computed and stored.

ADJUST - AZIMUTH - LEAST SQUARES method and ADJUST - BEARING - LEAST SQUARES method:

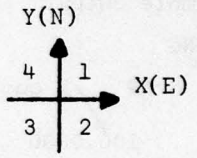
Data: ADJ/AZ/LSQ or [A/AL]

```
J  N
K  D1  A
L  D2  B
M  D3  C
N  D4  D
```

Data: ADJ/BR/LSQ or [A/BL]

```
...
...
```

Same as ADJ/ /COM except that the LSQ method is used instead of the COM method.



Bearing Quadrants Enter: Code, DEG, MIN, SEC

Example output for ADJ/AZ/COM and ADJ/AZ/LSQ:

J → 3 3 ← N

		Y	X	
K →	4	100.0000	300.0000	} coordinates calculated from input data
L →	1	300.0000	300.0000	
M →	2	300.0000	100.0000	
N →	3	100.0305	103.4905	

Y ERROR= -0.0305 X ERROR= -3.4905 TOT. ERR= 3.4906

AZ OF C.L. = 269 29 58.8 PERIM= 800.0000 RATIO= 1/229.19 ← one ft. of error for every 229.19 feet of perimeter

4	100.0000	298.2548	} coordinates of adjusted points
1	299.9543	298.2544	
2	299.9547	96.5244	

PT	TO	PT	DISTANCE	AZIMUTH		
3		4	198.2548	90	0	0.0
4		1	199.9543	359	59	59.6
1		2	201.7300	270	0	0.4
2		3	199.9849	179	0	15.1
3		3	0.	90	0	0.0

back → 3 ← ahead

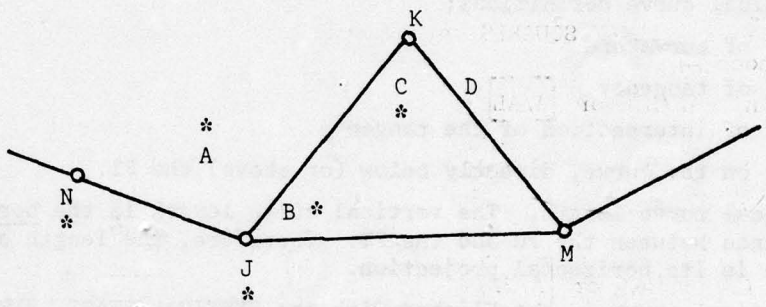
0 1 11

Data: TRIANGULATION or [TRI]

N J A B K C M C

Known: N J A B C

Repeated use of this command can be used to compute a network of simple explicit triangulation. It may also be used in conjunction with the other commands to compute a network of combined triangulation and traverses.



*=known values

Example output:

```
TRIANGULATIO                                  °   '   "                °   '   "
N=       7    J=      6      A=    45    0    0.      B=     90    0    0.
K=      16    C=    45    0    0.      M=     17 D=          100.0000
K→PT.    16      Y=        70.7107      X=          200.0000
M→PT.    17      Y=        0.           X=          270.7107
          J→6   K→16          70.7107      0   0   0.0
            6     17          70.7107     90   0   0.0
          K→16    M→17        100.0000    135  0   0.0
                                          °   '   "
                              Dist
```

Vertical Curve Commands

Vertical curves are used to join the intersecting grades of railroads, highways, or other routes, and tend to smooth out changes in vertical motion.

The plane which contains a vertical curve has elevation as the ordinate and station as the abscissa.

Vertical curves are always parabolic (as compared with circular or spiral curves of the horizontal plane).

Vertical curve definitions:

PC Point of curvature.

PT Point of tangency.

PI Point of intersection of the tangents.

PIC Point on the curve, directly below (or above) the PI.

VCL Vertical curve length. The vertical curve length is the horizontal distance between the PC and the PT. Therefore, the length of the curve is its horizontal projection.

A vertical curve is established with the VERTICAL/START, VERTICAL/END sequence (see below). Once a curve has been established the following commands may be used:

EVEN/STATIONS

OFFSET/ELEVATION

CURVE/DRAIN

How to enter stations. In all the vertical commands, stations must be entered as decimal feet. For example, Station 75 + 44 would be entered as 7544. Vertical curve lengths also are entered in decimal feet.

Note: In the vertical curve section, points are defined by station and elevation rather than by coordinates. Do not use point numbers that are the same as used for any other command in the same run. A survey point that has vertical data also should be given another number for its vertical curve calculations.

Data: VERTICAL/STA or [V/S]

one such	N			
line for each curve except last one	I	S(I)	E(I)	VCL(I)
	J	S(J)	E(J)	VCL(J)
	K	S(K)	E(K)	VCL(K)
	L	S(L)	E(L)	VCL(L)

N= number of curves (max=10) in VERTICAL/STA command.

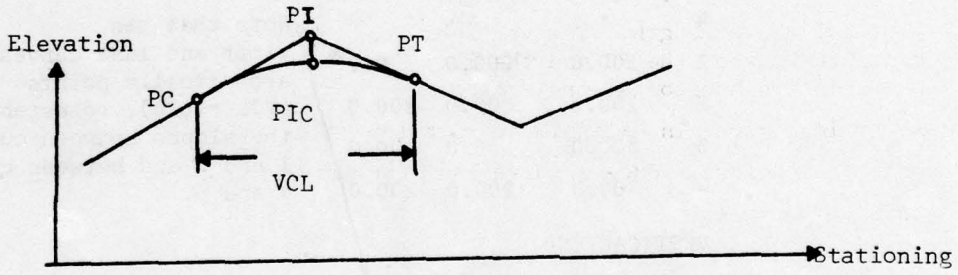
I,J,K,L,M = curve identification numbers.

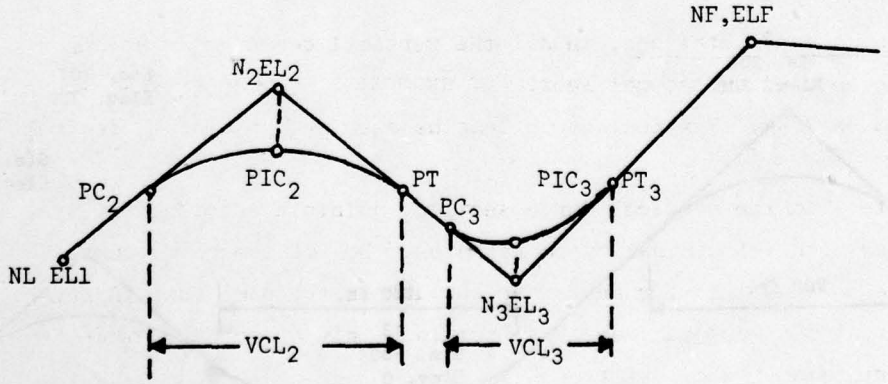
S = Station of point (feet).

E = Elevation of PI.

Data: VERTICAL/END or [V/E]

last curve	M	S(M)	E(M)	VCL(M)





The VERTICAL/STA and VERTICAL/END commands define a vertical alignment by specifying the station and elevation of all the tangent intersection points of the alignment. Vertical curve lengths (measured horizontally) are also specified. The first point must be specified by a VERTICAL/START command and the last point by a VERTICAL/END command. There can be no other commands between these two.

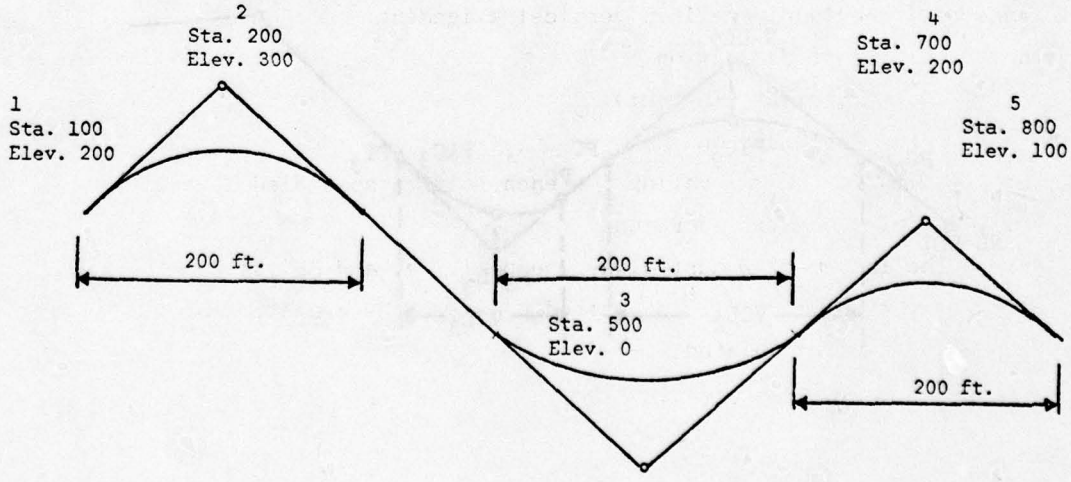
These commands must precede any other vertical curve commands for the particular alignment. Only one vertical alignment may be processed at one time.

A maximum of ten points may be specified, and output will be generated for each point after all points have been stored. Curve information is stored for the other vertical routines that follow.

Example data:

```
VERTICAL/STA
4
1     100.0     200.0     0.0
2     200.0     300.0   200.0
3     500.0       0.0   200.0
4     700.0     200.0   200.0
VERTICAL/END
5     800.0     100.0     0.0
```

Note that the first and last curves are actually points (VCL = 0.0), to establish the slopes between curves 1 and 2 and between curves 4 and 5.



```
VERTICAL/STA

VERTICAL/END

PT.     1     STA=          100.00000        ELEV=          200.00000

PT.     2     STA=          200.00000        ELEV=          300.00000
NPC   STA=      100.00000        ELEV=      200.00000
NPT   STA=      300.00000        ELEV=      200.00000
PIC   STA=      200.00000        ELEV=      250.00000
GRADE FROM PT.     1 TO PT.       2 =      100.00000

PT.     3     STA=          500.00000        ELEV=            0.
NPC   STA=      400.00000        ELEV=      100.00000
NPT   STA=      600.00000        ELEV=      100.00000
PIC   STA=      500.00000        ELEV=       50.00000
GRADE FROM PT.     2 TO PT.       3 =     -100.00000

PT.     4     STA=          700.00000        ELEV=          200.00000
NPC   STA=      600.00000        ELEV=      100.00000
NPT   STA=      800.00000        ELEV=      100.00000
PIC   STA=      700.00000        ELEV=      150.00000
GRADE FROM PT.     3 TO PT.       4 =      100.00000
PT.     5     STA=          800.00000        ELEV=          100.00000
GRADE FROM PT.     4 TO PT.       5 =     -100.00000
```

Data: EVEN/STATION or [E/S]

S1 S SF

This routine calculates elevations at stations S1 through SF at increments of D. The stations S1 through SF must be within the range of a previously defined vertical alignment.

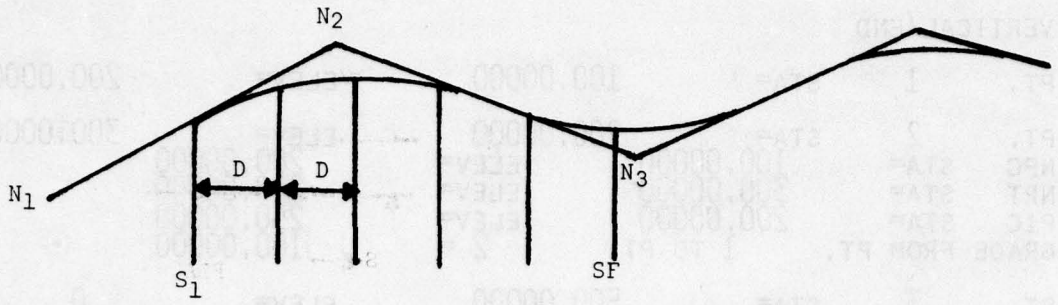
Input: S1 Starting station

D Increment (in feet)

SF Final station

Output: Station and elevation for each station specified by the station limits and the increment.

Note: The increment D must always be positive, and SF must be greater than S1. If D is entered as zero, the elevation of S1 only will be calculated.



Example data: EVEN/STATION
200.0 50.0 500.0

Example output:

EVEN/STATION

STA=	200.00000	ELEV=	250.00000
STA=	250.00000	ELEV=	237.00000
STA=	300.00000	ELEV=	200.00000
STA=	350.00000	ELEV=	150.00000
STA=	400.00000	ELEV=	100.00000
STA=	450.00000	ELEV=	62.50000
STA=	500.00000	ELEV=	50.00000

Data: OFFSET/ELEVA or [O/E]

N S D G

This routine calculates the elevation of point N, using the following input information:

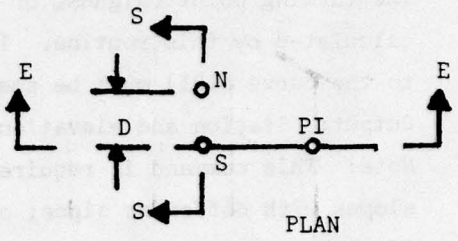
N Number of the required point

S Station of the point

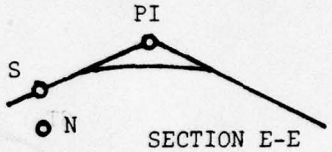
D Horizontal distance off the alignment at the station

G Percentage grade from the horizontal, at right angles to the center line. (Negative means down)

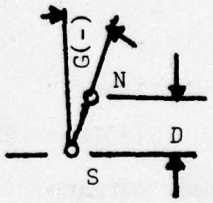
Output: Station and elevation of N.



PLAN



SECTION E-E



SECTION S-S

Example output:

OFFSET/ELEVA

PT. 8 STA= 150.00000 ELEV= 238.00000

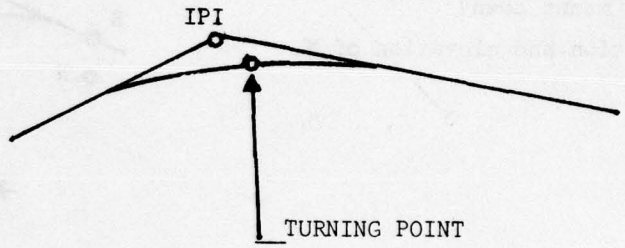
Data: CURVE/DRAIN or [C/D]

IPI

The turning point (highest or lowest point) of a vertical curve is calculated by this routine. The point of intersection of the tangents to the curve (IPI) must be specified in the input.

Output: Station and elevation of the turning point.

Note: This command is required only when the two tangents have slopes with different signs; otherwise, there is no turning point. If the two slopes are not of different signs, this command will produce erroneous results.



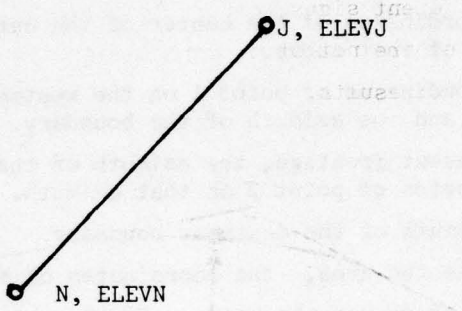
Example output:

```
CURVE/DRAIN
PT.    2     STA=     200.00000     ELEV=     250.00000
```

Data: SLOPE/LENGTH or [S/L]

N ELEVN J ELEVJ

This routine uses the known coordinates and elevation of points N and J to find the slope length of the tangent (distance) between the two points. The coordinates must have been calculated previously in the run. This command is not associated with the vertical curve commands.



Output: Slope length of N - J.

Example Output:

```
SLOPE/LENGTH
FROM PT.      6 TO PT.      7 SLOPE LENGTH=      141.42136
```

EXAMPLE I

A parcel of land is being changed because of the construction of a new street of expressway. A circular curve is going to be cut through the front of the existing property. The problem is to determine both the area that is being removed from the property owner and the frontage length. Referring to the sketch, the information that is known is:

1. The coordinates of the center of the curve (point 3) and the radius of that curve.
2. The coordinates of point 1 on the westmost boundary of the parcel and the azimuth of the boundary.
3. The present frontage, the azimuth of that frontage, and the coordinates of point 2 on that azimuth.
4. The azimuth of the eastmost boundary.

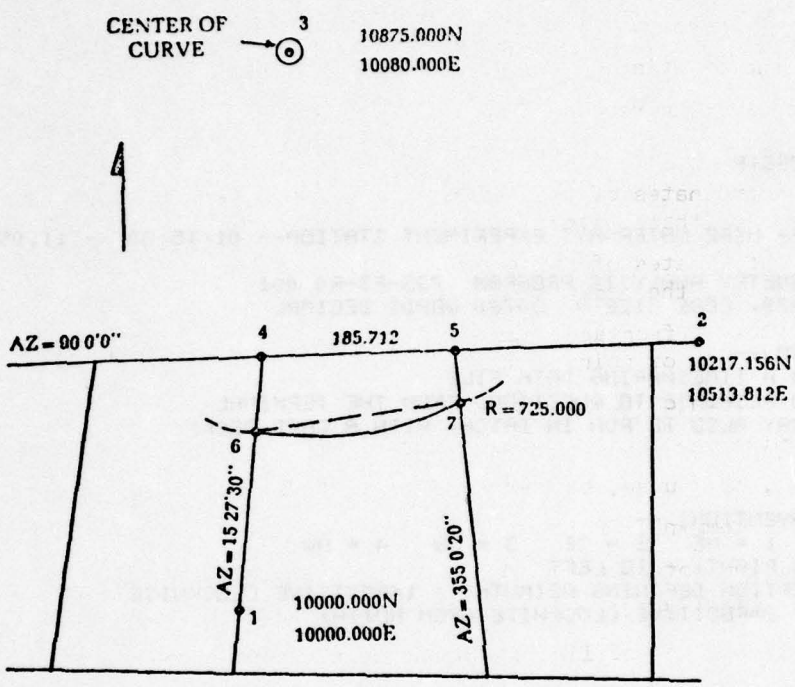
To determine the area, the coordinates of the boundary points of that area must, of course, be known. First, the known points are stored using the STORE command. Point 4 is determined by finding the coordinates of the intersection of the two lines meeting at point 4. The command is AZIMUTH/INT, which determines these coordinates from point 1 and the azimuth of line 1 - 4 and point 2 and the azimuth of line 2 - 4.

Point 5 is determined by locating it 185.712 feet and at an azimuth of 90° 0' 0" from point 4. The LOCATE/AZI command is used in this case.

Both points 6 and 7 can be found by intersecting the curve with the lines formed by the east and west boundaries. If these lines are defined by points and azimuths, as these are in this case, the command is ARC/LINE/AZ.

Now the area can be determined by first using the AREA command to determine the area of the trapezoid 6 4 5 7 6 and then using the SEGMENT/PL command to add to that area the segment area. The frontage (arc length) is automatically obtained as a by-product of this command.

A sketch of the relevant geometry, the data file solving the problem and the computer output is given below.



LIST MANDATA

```
10 CLEAR
20 1,10
30 STORE
40 3
50 1,10000.,10000.
60 2,10217.156,10513.812
70 3,10875,10080
80 AZIMUTH/INT
90 4,1,15,27,30,2,90,0,0
110 LOCATE/AZI
120 4,5,185.712,90,0,0
130 ARC/LINE/AZ
140 6,3,725,1,15,27,30,1
160 ARC/LINE/AZ
170 7,3,725,5,355,0,20,1
190 AREA
195 5
200 6,4,5,7,6
210 SEGMENT/PL
220 6,7,725
230 HDCOPY
240 ERASE
250 PAUSE

ready

*
```

```
RUN WESLIB/GCOGO,R


PROGRAM COGO -- USAE WATERWAYS EXPERIMENT STATION-- 01/15/80 -- 11.055

COORDINATE GEOMETRY ANALYSIS PROGRAM  733-F3-R0 001
REVISED AUG 1979, CORE SIZE =  30720 WORDS DECIMAL

DATA INPUT FORM --
 ENTER 0 IF IN A TIMESHARING DATA FILE
    OR 1 IF IN RESPONSE TO QUESTIONS FROM THE TERMINAL
         (IT MAY ALSO BE RUN IN BATCH, WITH A CARD DECK)
=0


ANGLE DATA CONVENTIONS --
   QUADRANTS   1 = NE   2 = SE   3 = SW   4 = NW
   SIGN = + TO RIGHT, - TO LEFT
   SELECT AN OPTION DEFINING AZIMUTHS ( 1=POSITIVE CLOCKWISE
   FROM SOUTH, 2=POSITIVE CLOCKWISE FROM NORTH)

=2

ENTER THE FILE DESCRIPTION OF YOUR DATA FILE
=MANDATA


CLEAR
BETWEEN POINTS     1  AND    10


STORE


AZIMUTH/INT

PT.     4     Y=     10217.1560     X=     10060.0526


LOCATE/AZI

    4     5       185.7120    90     0    0.
PT.     5     Y=     10217.1560     X=     10245.7646


ARC/LINE/AZ

PT.     6     Y=     10151.0092     X=     10041.7604
```

```
ARC/LINE/AZ

PT.      7      Y=      10170.1813      X=      10249.8698


AREA

   6     4     5     7     6

SELECT AN OPTION (1=ERASE, 2=COPY THEN ERASE)
=1

     11069.4115 SQFT.          0.2541 ACRES


SEGMENT/PL

THROUGH POINTS     6 AND     7    RADIUS =   725.00000
CHORD=         208.9906      ARC=         209.7211
CENTRAL ANGLE =     16     34      26.3
SEGMENT AREA=      1055.81842 SQFT.          0.02424 ACRES
NET AREA =     12125.22998 SQ. FT. =          0.27836 ACRES


HDCOPY



ERASE



PAUSE
INPUT 0 TO READ FROM T.S. FILE OR 1 TO READ FROM TERMINAL
=1



NEXT COMMAND WORD
=END
DO YOU HAVE MORE DATA TO RUN?
(0 NO, 1 YES )
=0

◆
```

EXAMPLE II

Listed below is a fairly lengthy data file which is stored in WESLIB as WESLIB/D.COGO,R. These data test most of the COGO commands without using graphics. The numerical results corresponding to these data are also given.

```
LIST D.COGO

1000 COMMENT
1010 TEST DATA FILE 'COGODATA' FOR PROGRAM COGO ON WES G-635 COMPUTER
1020 SKIP
1025 3
1030 COMMENT
1040 DATA SERIES FOR HORIZONTAL CURVE COMMANDS
1050 CLEAR
1060 1.   20.
1070 STORE
1080 5.
1090 1.   600.   200.
1100 4.   600.   700.
1110 6.   100.   700.
1120 10.   200.   701.
1130 11.   600.   700.
1140 ALIG
1150 589.   1.   4.   6.   2.   3.   5.   0.   0.   0.   100.
1160 COPOA
1170 8.   200.
1180 COOFF
1190 9.   200.   100.
1200 STAFC
1210 10.
1220 OFSA
1230 12.   11.
1240 SKIP
1245 3
1250 COMMENT
1260 RERUN HORIZ. CURVE COMMANDS, USING DEFCV COMMAND INSTEAD OF ALIG
1270 CLEAR
1280 1   20
1290 STORE
1300 8
1310 1   600.0   200.0
1320 2   600.0   300.0
1330 3   200.0   300.0
1340 4   600.0   700.0
1350 5   200.0   700.0
1360 6   100.0   700.0
1370 10 200.0   701.0
1380 11 600.0   700.0
1390 DEFCV
1400 599   2   100.0   4   5   728.3185   3   1
1430 COPOA
1440 8.   200.
1450 COOFF
1460 9.   200.   100.
1470 STAFC
1480 10.
1490 OFSA
1500 12.   11.
1550 STORE
1560 2
```

```
1570 22  200.0  100.0
1580 24  400.0  300.0
1600 DVLIN
1610 22  24  2
2000 SKIP
2005 3
2010 COMMENT
2020 START NEW SERIES FOR GIRDL COMMAND
2030 CLEAR
2040 1.  10.
2050 STORE
2060 5.
2070 1.  2.  1.
2080 2.  3.  2.
2090 3.  0.  5.
2100 4.  2.  7.
2110 5.  1.  8.
2120 GIRDL
2130 1.  4.  3.  2.  5.
2140 SKIP
2145 3
2150 COMMENT
2160 START NEW SERIES FOR GENERAL COMMANDS
2170 CLEAR
2180 1.  30.
2190 STORE
2200 5.
2210 1.  400.  600.
2220 2.  200.  800.
2230 3.  100.  600.
2240 6.  0.  200.
2250 8.  400.  200.
2260 ANGLE
2270 6.  8.  1.
2280 BEARING/INT
2290 7  8  3  45 0 0.00  6  4  45 0 0.00
2300 DISTANCE
2310 7.  8.
2320 INVERSE/AZ
2330 7.  8.
2340 INVERSE/BE
2350 7.  8.
2360 LOCATE/ANG
2370 6.  7.  10.  200.  -135.  0.  0.
2380 LOCATE/BEA
2390 6.  11.  100.  1.  90.  0.  0.
2400 LOCATE/LIN
2410 6.  11.  5.  400.
2420 ARC/LINE/PT
2430 9.  6.  200.  11.  5.  11.
2440 LOCATE/DEF
2450 6.  5.  4.  100.  -90.  0.  0.
2460 PARALLEL/L
2470 6.  8.  300.  12.  13.
2480 POINTS/INT
2490 14.  6.  1.  8.  5.
```

```
2500 ARC/ARC/INT
2510 21.  11.  141.42  12.  141.42  14.
2520 TANGENT/OFF
2530 15.  2.  4.  1.
2540 TRIANGULATIO
2550 7  6  45 0 0.00  90 0 0.00  16  45 0 0.00  17  100.0
2560 TANGENT
2570 18.  14.  100.  19.  15.  50.  1.  1.
2580 FORSECTION
2590 20.  7.  -45.  0.  0.  6.  45  0.  0.
2600 DUMP
2610 1.  20.
2620 REDEFINE
2630 17.  9.
2640 SEGMENT
2650 11  21  100.0
2660 AREA/AZIMUTH
2670 5
2680 15  1  8  20  15
2690 SEGMENT/PL
2700 11  21  100.0
2710 SEGMENT/MI
2720 11.  21.  100
2730 ARC/LINE/BR
2740 22.  20.  100.  14.  4.  90.  0.  0.  16.
2750 STORE
2760 1.
2770 24.  400.  300.
2800 DUMP
2810 1  25
2820 SAVE
2830 COGOSAVE
2840 1 10
2850 CLEAR
2860 1 10
2870 RESTORE
2880 COGOSAVE
2890 1 10
2900 COMMENT
2910 *** REMEMBER TO PURGE FILE COGOSAVE AFTER RUNNING THIS DATA ***
2920 DUMP
2930 1 10
2940 CLEAR
2950 1.  10.
2960 STORE
2970 3
2980 1 10000 10000
2990 2 10217.156 10513.812
3000 3 10875 10080
3010 AZIMUTH/INT
3020 4 1 15 27 30 2 90 0 0
3030 LOCATE/AZI
3040 4 5 185.712 90 0 0
3050 ARC/LINE/AZ
3060 6 3 725 1 15 27 30 1
3070 ARC/LINE/AZ
```

```
3080 7 3 725 5 355 0 20 1
3090 AREA/BEARING
3100 5
3110 6 4 5 7 6
3120 AREA
3130 5
3140 6 4 5 7 6
3150 SEGMENT/PL
3160 6 7 725
3170 SKIP
3175 3
3180 COMMENT
3190 START NEW SERIES FOR TRAVERSE ADJUSTMENT COMMANDS
3192 COMMENT
3194 ORDER OF TRAVERSE POINTS:  3 - 4 - 1 - 2 - CLOSE ON 3
3200 CLEAR
3210 1  20
3220 STORE
3230 1
3240 3  100.  100.
3250 ADJ/AZ/LSQ
3260 3  3
3270 4  200.  90.  0.  0.
3280 1  200.  0.  0.  0.
3290 2  200.  270.  0.  0.
3300 3  200.  179.  0.  0.
3310 ADJ/AZ/COM
3320 3  3
3330 4  200.  90.  0.  0.
3340 1  200.  0.  0.  0.
3350 2  200.  270.  0.  0.
3360 3  200.  179.  0.  0.
3370 CLEAR
3380 1  2
3382 CLEAR
3384 4  7
3390 ADJ/BR/LSQ
3400 3  3
3410 4  200.  1  90  0  0
3420 1  200.  4  0  0  0
3430 2  200.  4  90  0  0
3440 3  200.  3  1  0  0
3450 ADJ/BR/COM
3460 3  3
3470 4  200.  1  90  0  0
3480 1  200.  4  0  0  0
3490 2  200.  4  90  0  0
3500 3  200.  3  1  0  0
3510 CLEAR
3520 1  2
3524 CLEAR
3526 4  7
3530 ADJUST/ANG
3540 3  3
3550 3  4  200.  90.  0.  0.
3560 1  200.  90.  0.  0.
```

```
3570 2  200.   90.   0.   0.
3580 3  200.   90.   0.   0.
3590 SKIP
3595 3
3600 COMMENT
3610 START NEW SERIES FOR VERTICAL CURVE COMMANDS
3620 CLEAR
3630 1.   20.
3640 STORE
3650 2.
3660 6.   200.   100.
3670 7.   300.   100.
3680 VERTICAL/STA
3690 4.
3700 1.   100.   200.   0.
3710 2.   200.   300.   200.
3720 3.   500.   0.   200.
3730 4.   700.   200.   200.
3740 VERTICAL/END
3750 5.   800.   100.   0.
3760 EVEN/STATION
3770 200.   50.   500.
3780 CURVE/DRAIN
3790 2.
3800 SLOPE/LENGTH
3810 6.   200.   7.   100.
3820 OFFSET/ELEVA
3830 8.   150.   100.   .5
3840 SKIP
3845 3
3850 COMMENT
3860 END OF COGODATA
3870 COMMENT
3890 REMEMBER TO RELEASE THE FILE COGOSAVE THAT THIS RUN CREATED
3900 STOP
3910 0

ready

*
```

```
RUN COGOHSTR



PROGRAM COGO -- USAE WATERWAYS EXPERIMENT STATION-- 01/15/80 --  9.768

COORDINATE GEOMETRY ANALYSIS PROGRAM  733-F3-R0 001
REVISED AUG 1979, CORE SIZE =  30720 WORDS DECIMAL

DATA INPUT FORM --
 ENTER 0 IF IN A TIMESHARING DATA FILE
    OR 1 IF IN RESPONSE TO QUESTIONS FROM THE TERMINAL
         (IT MAY ALSO BE RUN IN BATCH, WITH A CARD DECK)
=0


ANGLE DATA CONVENTIONS --
   QUADRANTS   1 = NE   2 = SE   3 = SW   4 = NW
   SIGN = + TO RIGHT, - TO LEFT
   SELECT AN OPTION DEFINING AZIMUTHS ( 1=POSITIVE CLOCKWISE
   FROM SOUTH, 2=POSITIVE CLOCKWISE FROM NORTH)

=2

ENTER THE FILE DESCRIPTION OF YOUR DATA FILE
=D.COGO


COMMENT
1010 TEST DATA FILE 'COGODATA' FOR PROGRAM COGO ON WES G-635 COMPUTER


SKIP



COMMENT
1040 DATA SERIES FOR HORIZONTAL CURVE COMMANDS


CLEAR
BETWEEN POINTS     1  AND    20


STORE


ALIG
ALIGNMENT

  589 ( 1.)  R=    400.0000   T=    400.0000   ANGLE   90    0    0.0
```

```
LONG CORD=     565.6854    MID ORD=     117.1573    EXT DIST=     165.6854
X= 100.00000  SPC=     100.0000  ARC LNG=     628.3185  SPT=     728.3185
POC    2    Y=         600.0000    X=         300.0000       SPI=     500.0000
POT    5    Y=         200.0000    X=         700.0000  DEG=    14   19   26.2
CTR    3    Y=         200.0000    X=         300.0000
POI    4    Y=         600.0000    X=         700.0000


COPOA

PT.      8      Y=         587.5650      X=         398.9616


COOFF

PT.      9      Y=         490.6737      X=         374.2212


STAFC

   10         728.3185


OFSA
OFSAL

PT     12  PT     11  STA.      414.16  OFFSET       -165.6854


SKIP



COMMENT
1260 RERUN HORIZ. CURVE COMMANDS, USING DEFCV COMMAND INSTEAD OF ALIG


CLEAR
BETWEEN POINTS      1  AND     20


STORE


DEFCV

CURVE NO.   599      SIGN =  1        (DEFINED)
PC POINT      2     STA PC =     100.0000          PI POINT     4
PT POINT      5     STA PT =     728.3185          CENTER POINT     3


COPOA

PT.      8      Y=         587.5650      X=         398.9616
```

```
COOFF

PT.      9      Y=          490.6737      X=          374.2212


STAFC

    10          728.3185


OFSA
OFSAL

PT     12  PT     11  STA.      414.16  OFFSET        -165.6854


STORE


DVLIN

        141.4214
PT.      22       Y=          200.0000      X=          100.0000
STATION     -100.0000       OFFSET=           400.0000
PT.      23       Y=          300.0000      X=          200.0000
STATION        0.0000       OFFSET=           300.0000
PT.      24       Y=          400.0000      X=          300.0000
STATION      100.0000       OFFSET=           200.0000


SKIP


COMMENT
2020 START NEW SERIES FOR GIRDL COMMAND


CLEAR
BETWEEN POINTS      1   AND     10


STORE


GIRDL

    PT  TO PT        DISTANCE
     1       2          1.4142
     2       3          4.2426
     3       4          2.8284
     4       5          1.4142
```

```
SKIP



COMMENT
2160 START NEW SERIES FOR GENERAL COMMANDS

CLEAR
BETWEEN POINTS      1  AND     30

STORE

ANGLE

CLOCKWISE ANGLE FROM     6, AROUND     8, TO     1, =  270    0    0.

BEARING/INT

     7      8      3     45      0     0.       6      4     45      0     0.
PT.      7      Y=         200.0000      X=          -0.0000

DISTANCE

FROM PT.    7 TO PT.    8  DIST=         282.8427

INVERSE/AZ

FROM     7 TO     8, DIST =         282.8427
AZIMUTH =   45    0    0.0

INVERSE/BE

FROM       7 TO     8, DIST =         282.8427
BEARING =   45    0    0.0 IN QUADRANT  1

LOCATE/ANG

     6      7     10         200.0000       -135    0     0.
PT.      10      Y=         400.0000      X=           0.

LOCATE/BEA

     6     11             100.00000          1         90    0     0.
```

```
PT.     11      Y=          0.          X=        300.0000

LOCATE/LIN

     6    11     5        400.0000
PT.      5      Y=          0.          X=        600.0000

ARC/LINE/PT

PT.      9      Y=          0.          X=        400.0000

LOCATE/DEF

     6     5     4         100.0000       -90    0     0.
PT.      4      Y=        100.0000      X=        600.0000

PARALLEL/L

     6     8        300.0000     12     13
PT.      12      Y=          0.          X=        500.0000
PT.      13      Y=        400.0000      X=        500.0000

POINTS/INT

    14     6     1     8     5
PT.      14      Y=        200.0000      X=        400.0000

ARC/ARC/INT

    21    11        141.4200     12        141.4200     14
PT.      21      Y=         99.9981      X=        400.0000

TANGENT/OFF

PT.     15      Y=        200.0000      X=        600.0000
FROM PT.    4 TO PT.   15    DIST=        100.0000
FROM PT.    2 TO PT.   15    DIST=        200.0000

TRIANGULATIO

N=      7    J=      6    A=   45    0     0.     B=   90    0     0.
K=     16    C=   45    0     0.     M=     17   D=          100.0000
PT.      16      Y=         70.7107      X=        200.0000
PT.      17      Y=          0.          X=        270.7107
      6     16         70.7107          0      0      0.
      6     17         70.7107         90      0      0.
     16     17        100.0000        135      0      0.
```

```
TANGENT

PT.     18      Y=          296.8246      X=          425.0000
PT.     19      Y=          248.4123      X=          612.5000
    18     19          193.6492      1     49    24.5


FORSECTION

    20      7  -45.0     0.       0.        6   45.0     0.       0.
        200.0000          200.0000          282.8427
PT.     20      Y=          200.0000      X=          200.0000


DUMP

 POINT          Y                X
    1        400.0000         600.0000
    2        200.0000         800.0000
    3        100.0000         600.0000
    4        100.0000         600.0000
    5          0.             600.0000
    6          0.             200.0000
    7        200.0000          -0.0000
    8        400.0000         200.0000
    9          0.             400.0000
   10        400.0000           0.
   11          0.             300.0000
   12          0.             500.0000
   13        400.0000         500.0000
   14        200.0000         400.0000
   15        200.0000         600.0000
   16         70.7107         200.0000
   17          0.             270.7107
   18        296.8246         425.0000
   19        248.4123         612.5000
   20        200.0000         200.0000


REDEFINE
POINT    17 IS NEW LOCATION FOR POINT     9


SEGMENT

THROUGH POINTS    11 AND    21    RADIUS =   100.00000
CHORD=          141.4200       ARC=          157.0777
CENTRAL ANGLE =     89     59      56.1
SEGMENT AREA=       2853.88586 SQFT.          0.06552 ACRES


AREA/AZIMUTH

    15      1          200.0000      0      0     0.
```

```
  1     8        400.0000   270      0     0.
  8    20        200.0000   180      0     0.
 20    15        400.0000    90      0     0.



    79999.9990 SQFT.          1.8365 ACRES

SEGMENT/PL

THROUGH POINTS   11 AND   21   RADIUS =  100.00000
CHORD=        141.4200      ARC=        157.0777
CENTRAL ANGLE =     89     59      56.1
SEGMENT AREA=      2853.88586 SQFT.          0.06552 ACRES
NET AREA =     82853.88477 SQ. FT. =          1.90206 ACRES

SEGMENT/MI

THROUGH POINTS   11 AND   21   RADIUS =  100.00000
CHORD=        141.4200      ARC=        157.0777
CENTRAL ANGLE =     89     59      56.1
SEGMENT AREA=     -2853.88586 SQFT.         -0.06552 ACRES
NET AREA =     79999.99902 SQ. FT. =          1.83655 ACRES

ARC/LINE/BR

PT.    22      Y=        200.0000      X=        100.0000

STORE

DUMP

 POINT          Y              X
    1        400.0000       600.0000
    2        200.0000       800.0000
    3        100.0000       600.0000
    4        100.0000       600.0000
    5          0.           600.0000
    6          0.           200.0000
    7        200.0000        -0.0000
    8        400.0000       200.0000
    9          0.           270.7107
   10        400.0000         0.
   11          0.           300.0000
   12          0.           500.0000
   13        400.0000       500.0000
   14        200.0000       400.0000
```

```
   15        200.0000          600.0000
   16         70.7107          200.0000
   17           0.             270.7107
   18        296.8246          425.0000
   19        248.4123          612.5000
   20        200.0000          200.0000
   21         99.9981          400.0000
   22        200.0000          100.0000
   24        400.0000          300.0000

SAVE
     INTO FILE COGOSAVE
BETWEEN POINTS      1  AND    10

CLEAR
BETWEEN POINTS      1  AND    10

RESTORE
     FROM FILE COGOSAVE
BETWEEN POINTS      1  AND    10

COMMENT
2910 *** REMEMBER TO PURGE FILE COGOSAVE AFTER RUNNING THIS DATA ***

DUMP

 POINT         Y                 X
    1        400.0000          600.0000
    2        200.0000          800.0000
    3        100.0000          600.0000
    4        100.0000          600.0000
    5          0.              600.0000
    6          0.              200.0000
    7        200.0000            0.
    8        400.0000          200.0000
    9          0.              270.7107
   10        400.0000            0.

CLEAR
BETWEEN POINTS      1  AND    10

STORE

AZIMUTH/INT

PT.      4      Y=      10217.1560      X=      10060.0526
```

```
LOCATE/AZI

     4      5          185.7120     90      0     0.
PT.       5       Y=      10217.1560       X=       10245.7646

ARC/LINE/AZ

PT.       6       Y=      10151.0092       X=       10041.7603

ARC/LINE/AZ

PT.       7       Y=      10170.1814       X=       10249.8696

AREA/BEARING

     6      4          68.6296      1     15     27    30.0
     4      5         185.7120      1     90      0     0.
     5      7          47.1536      2      4     59    39.2
     7      6         208.9906      3     84     44    11.2



     11069.3861 SQFT.              0.2541 ACRES

AREA

    6    4    5    7    6
      11069.3861 SQFT.              0.2541 ACRES

SEGMENT/PL

THROUGH POINTS     6 AND     7    RADIUS =   725.00000
CHORD=         208.9906       ARC=         209.7211
CENTRAL ANGLE =      16     34      26.3
SEGMENT AREA=       1055.81920 SQFT.           0.02424 ACRES
NET AREA =     12125.20532 SQ. FT. =            0.27836 ACRES

SKIP



COMMENT
3190 START NEW SERIES FOR TRAVERSE ADJUSTMENT COMMANDS
```

```
COMMENT
3194 ORDER OF TRAVERSE POINTS:  3 - 4 - 1 - 2 - CLOSE ON 3


CLEAR
BETWEEN POINTS     1  AND    20


STORE


ADJ/AZ/LSQ

    3     3
    4         100.0000         300.0000
    1         300.0000         300.0000
    2         300.0000         100.0000
    3         100.0305         103.4905
 Y ERROR=      -0.0305 X ERROR=      -3.4905 TOT. ERR=       3.4906
AZ OF C.L. =   269  29   58.9  PERIM=       800.000  RATIO=1/  229.19
    4         100.0000         298.2548
    1         299.9543         298.2544
    2         299.9547          96.5244
   PT TO PT      DISTANCE        AZIMUTH
    3     4        198.2548     90     0     0.0
    4     1        199.9543    359    59    59.6
    1     2        201.7300    270     0     0.4
    2     3        199.9849    179     0    15.1
    3     3          0.         90     0     0.


ADJ/AZ/COM

    3     3
    4         100.0000         300.0000
    1         300.0000         300.0000
    2         300.0000         100.0000
    3         100.0305         103.4905
 Y ERROR=      -0.0305 X ERROR=      -3.4905 TOT. ERR=       3.4906
AZ OF C.L. =   269  29   58.9  PERIM=       800.000  RATIO=1/  229.19
    4          99.9924         299.1274
    1         299.9848         298.2547
    2         299.9772          97.3897
   PT TO PT      DISTANCE        AZIMUTH
    3     4        199.1274     90     0     7.9
    4     1        199.9943    359    45     0.0
    1     2        200.8650    269    59    52.3
    2     3        199.9943    179    15     7.8
    3     3          0.         90     0     0.


CLEAR
BETWEEN POINTS     1  AND     2
```

```
CLEAR
BETWEEN POINTS      4  AND      7


ADJ/BR/LSQ

     3      3
     4          100.0000          300.0000
     1          300.0000          300.0000
     2          300.0000          100.0000
     3          100.0304           96.5095
 Y ERROR=       -0.0304 X ERROR=        3.4905 TOT. ERR=        3.4906
AZ OF C.L. =    90  29   58.9  PERIM=         800.000  RATIO=1/  229.19
     4          100.0000          301.7452
     1          299.9543          301.7456
     2          299.9539          103.5061
   PT TO PT       DISTANCE          BEARING
     3      4       201.7452     90     0     0.0  SE
     4      1       199.9543      0     0     0.4  NE
     1      2       198.2395     89    59    59.6  SW
     2      3       199.9846      1     0    16.4  SW
     3      3         0.         90     0     0.    SE


ADJ/BR/COM

     3      3
     4          100.0000          300.0000
     1          300.0000          300.0000
     2          300.0000          100.0000
     3          100.0304           96.5095
 Y ERROR=       -0.0304 X ERROR=        3.4905 TOT. ERR=        3.4906
AZ OF C.L. =    90  29   58.9  PERIM=         800.000  RATIO=1/  229.19
     4           99.9924          300.8726
     1          299.9848          301.7453
     2          299.9771          102.6255
   PT TO P      DISTANCE          BEARING
     3      4       200.8726     89    59    52.2  SE
     4      1       199.9943      0    15     0.0  NE
     1      2       199.1198     89    59    52.0  SW
     2      3       199.9943      0    45     7.9  SW
     3      3         0.         90     0     0.    SE


CLEAR
BETWEEN POINTS      1  AND      2


CLEAR
BETWEEN POINTS      4  AND      7


ADJUST/ANG

     3      3
     4          100.0000          300.0000
```

```
    1        300.0000        300.0000
    2        300.0000        100.0000
    3        100.0000        100.0000
 Y ERROR=      -0.0000 X ERROR=       0.0000 TOT. ERR=       0.0000
AZ OF C.L. =  138   0   46.0  PERIM=       800.000  RATIO=1/ 0.31176E 08
    4        100.0000        300.0000
    1        300.0000        300.0000
    2        300.0000        100.0000
   PT TO PT      DISTANCE         ANGLE
    3     4        200.0000     90     0     0.0
    4     1        200.0000      0     0     0.0
    1     2        200.0000    270     0     0.0
    2     3        200.0000    180     0     0.0
    3     3          0.         90     0     0.

SKIP



COMMENT
3610 START NEW SERIES FOR VERTICAL CURVE COMMANDS

CLEAR
BETWEEN POINTS     1  AND    20

STORE

VERTICAL/STA


VERTICAL/END

PT.      1     STA=       100.00000     ELEV=       200.00000

PT.      2     STA=       200.00000     ELEV=       300.00000
NPC   STA=        100.00000      ELEV=        200.00000
NPT   STA=        300.00000      ELEV=        200.00000
PIC   STA=        200.00000      ELEV=        250.00000
GRADE FROM PT.      1 TO PT.       2 =        100.00000

PT.      3     STA=       500.00000     ELEV=          0.
NPC   STA=        400.00000      ELEV=        100.00000
NPT   STA=        600.00000      ELEV=        100.00000
PIC   STA=        500.00000      ELEV=         50.00000
GRADE FROM PT.      2 TO PT.       3 =       -100.00000

PT.      4     STA=       700.00000     ELEV=       200.00000
NPC   STA=        600.00000      ELEV=        100.00000
NPT   STA=        800.00000      ELEV=        100.00000
```

```
PIC   STA=       700.00000      ELEV=       150.00000
GRADE FROM PT.       3 TO PT.       4 =       100.00000
PT.      5     STA=       800.00000      ELEV=       100.00000
GRADE FROM PT.       4 TO PT.       5 =      -100.00000

EVEN/STATION

STA=       200.00000      ELEV=       250.00000
STA=       250.00000      ELEV=       237.50000
STA=       300.00000      ELEV=       200.00000
STA=       350.00000      ELEV=       150.00000
STA=       400.00000      ELEV=       100.00000
STA=       450.00000      ELEV=        62.50000
STA=       500.00000      ELEV=        50.00000

CURVE/DRAIN

PT.      2      STA=       200.00000      ELEV=       250.00000

SLOPE/LENGTH

FROM PT.       6 TO PT.       7, SLOPE LENGTH =       141.42135

OFFSET/ELEVA

PT.      8      STA=       150.00000      ELEV=       238.00000

SKIP



COMMENT
3860 END OF COGODATA


COMMENT
3890 REMEMBER TO RELEASE THE FILE COGOSAVE THAT THIS RUN CREATED


STOP
END
DO YOU HAVE MORE DATA TO RUN?
(0 NO, 1 YES )
=0
```

EXAMPLE III

The following data case demonstrates the graphics capabilities developed for the Tektronix 4014 terminal. The terminal should be run using small size characters. To properly set the terminal, the user should switch to local mode, press ESC followed by ; , and then swith back to on line.

```
LIST GRAPHDAT

100 SHOW
110 0
120 CLEAR
130 1,50
140 STORE
150 10
160 1,500,50
170 2,500,300
180 3,50,300
190 4,50,600
200 20,200,0
210 25,500,0
220 30,100,100
230 35,100,500
240 49,0,0
250 50,550,650
260 ALIG
270 101,1,2,3,5,6,7,200,0,100,0
280 ALIG
290 102,7,3,4,8,9,10,150,0,-1,0
300 SHOW
310 1
320 WINDOW
330 0,0,0,0
340 NORTH/SCALE
350 0,0
360 LSTCRV
370 2
380 101,102
390 GETCRV
400 102
410 GETCRV
420 101
430 COPOA
440 40,250
450 COOFF
460 41,275,50
470 STAFC
480 40
490 OFSAL
500 42,50
510 DVLIN
520 20,25,4
530 HDCOPY
540 ERASE
550 ARC/ARC/INT
560 45,6,250,9,300,7
570 HDCOPY
580 ERASE
590 AREA/AZIMUTH
600 6
610 2,9,10,3,6,2
620 HDCOPY
```

```
630 ERASE
640 GETCRV
650 102
660 HDCOPY
670 ERASE
680 IVLIN
690 30,35,4
700 GIRDL
710 20,25,4,30,35
720 HDCOPY
730 ERASE
740 TANGENT/OFF
750 46,6,5,7
760 HDCOPY
770 ERASE
780 TANGENT
790 61,6,150,62,9,100,1,1
800 TANGENT
810 63,6,150,64,9,100,-1,1
820 TANGENT
830 65,6,150,66,9,100,1,-1
840 TANGENT
850 67,6,150,68,9,100,-1,-1
860 HDCOPY
870 ERASE
880 TRIANGULATIO
890 6,7,90,0,0,45,0,0,71,90,0,0,72,100
900 HDCOPY
910 ERASE
920 WINDOW
930 -50,50,400,850
940 VERTICAL/STA
950 4
960 1,100,200,0
970 2,200,300,200
980 3,500,0,200
990 4,700,200,200
1000 VERTICAL/END
1010 5,800,100,0
1020 HDCOPY
1030 ERASE
1040 COMMENT
1050 EXAMPLE PROBLEM INVOLVING SEVERAL CURVES
1060 CLEAR
1070 1,500
1080 STORE
1090 9
1100 2,4750,0
1110 4,2750,1550
1120 6,4900,5750
1130 8,2000,9050
1140 10,4050,12350
1150 12,8150,11100
1160 14,8100,4000
1170 15,3250,11150
```

```
1180 16,6250,2250
1190 DUMP
1200 1,50
1210 HDCOPY
1220 ERASE
1230 WINDOW
1240 0,0,0,0
1250 ALIG
1260 500,2,4,6,1,3,5,1700.,0.,2200.,0.
1270 ALIG
1280 501,5,6,8,9,7,11,0.,2000.,-1.,0.
1290 ALIG
1300 502,11,8,10,11,13,15,0.,0.,-1.,0.
1310 ALIG
1320 503,15,10,12,15,17,19,0.,0.,-1.,0.
1330 ALIG
1340 504,19,12,14,21,23,18,0.,0.,-1.,2000.
1350 ALIG
1360 505,18,14,16,19,21,20,3800.,0.,-1.,0.
1370 HDCOPY
1380 SHOW
1390 1
1400 WINDOW
1410 0,0,0,0
1420 ERASE
1430 SHOW/N-PTS
1440 0
1450 HDCOPY
1460 LINE/N-PTS
1470 8
1480 2,4,6,8,10,12,14,16
1490 NORTH/SCALE
1500 1000,1000
1510 HDCOPY
1520 GETCRV
1530 500
1540 GETCRV
1550 501
1560 GETCRV
1570 502
1580 GETCRV
1590 503
1600 GETCRV
1610 504
1620 GETCRV
1630 505
1640 HDCOPY
1650 ERASE
1660 COMMENT
1670 END OF DATAFILE HAS BEEN REACHED
1680 PAUSE

ready

*
```

```
PROGRAM COGO -- USAE WATERWAYS EXPERIMENT STATION-- 01/15/80 --  9.518

COORDINATE GEOMETRY ANALYSIS PROGRAM  733-F3-R0 001
REVISED AUG 1979. CORE SIZE =  30720 WORDS DECIMAL

DATA INPUT FORM --
 ENTER 0 IF IN A TIMESHARING DATA FILE
    OR 1 IF IN RESPONSE TO QUESTIONS FROM THE TERMINAL
         (IT MAY ALSO BE RUN IN BATCH. WITH A CARD DECK)
=0


ANGLE DATA CONVENTIONS --
   QUADRANTS   1 = NE   2 = SE   3 = SW   4 = NW
   SIGN = + TO RIGHT, - TO LEFT
   SELECT AN OPTION DEFINING AZIMUTHS ( 1=POSITIVE CLOCKWISE
   FROM SOUTH. 2=POSITIVE CLOCKWISE FROM NORTH)

=2

ENTER THE FILE DESCRIPTION OF YOUR DATA FILE
=GRAPHDAT


SHOW


CLEAR

BETWEEN POINTS     1  AND    50

STORE


ALIG
ALIGNMENT

  101 ( 1.)  R=    200.0000   T=    200.0000   ANGLE   90    0    0.0
LONG CORD=    282.8427   MID ORD=     58.5786   EXT DIST=     82.8427
X=  50.00001  SPC=     150.0000  ARC LNG=    314.1593  SPT=    464.1593
POC    5   Y=       500.0000   X=       100.0000       SPI=    350.0000
POT    7   Y=       300.0000   X=       300.0000   DEG=   28   38   52.4
CTR    6   Y=       300.0000   X=       100.0000
POI    2   Y=       500.0000   X=       300.0000


ALIG
ALIGNMENT

  102 (-1.)  R=    150.0000   T=    150.0000   ANGLE   90    0    0.
LONG CORD=    212.1320   MID ORD=     43.9340   EXT DIST=     62.1320
X= 100.00001  SPC=     564.1593  ARC LNG=    235.6194  SPT=    799.7787
POC    8   Y=       200.0000   X=       300.0000       SPI=    714.1593
POT   10   Y=        50.0000   X=       450.0000   DEG=   38   11   49.9
CTR    9   Y=       200.0000   X=       450.0000
POI    3   Y=        50.0000   X=       300.0000


SHOW
SELECT AN OPTION (1=ERASE, 2=COPY THEN ERASE)
=2
```

WINDOW

NORTH/SCALE

LSTCRV

TABLE OF CURVES

CURVE	NB	NPI	NA	NPC	NC	NPI	R	T	SP	X
101	1	2	3	5	6	7	200.00	0.	100.00	0.

SETCRV

SETCRV

COPCA

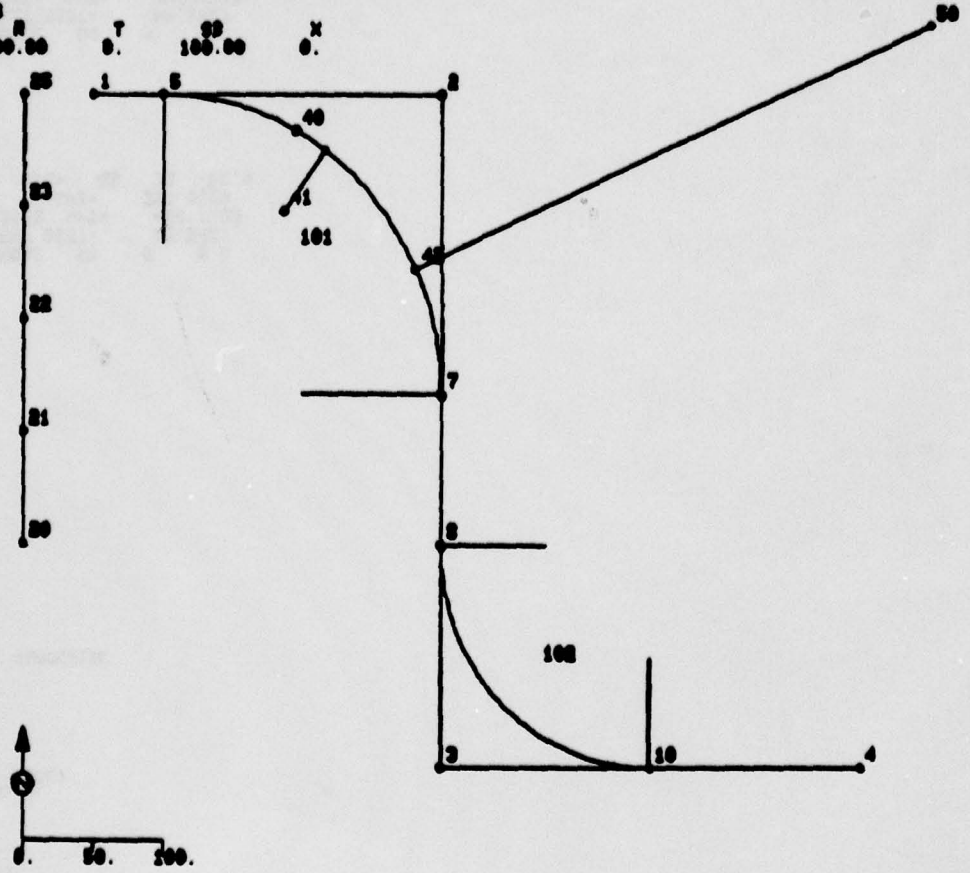
COOFF

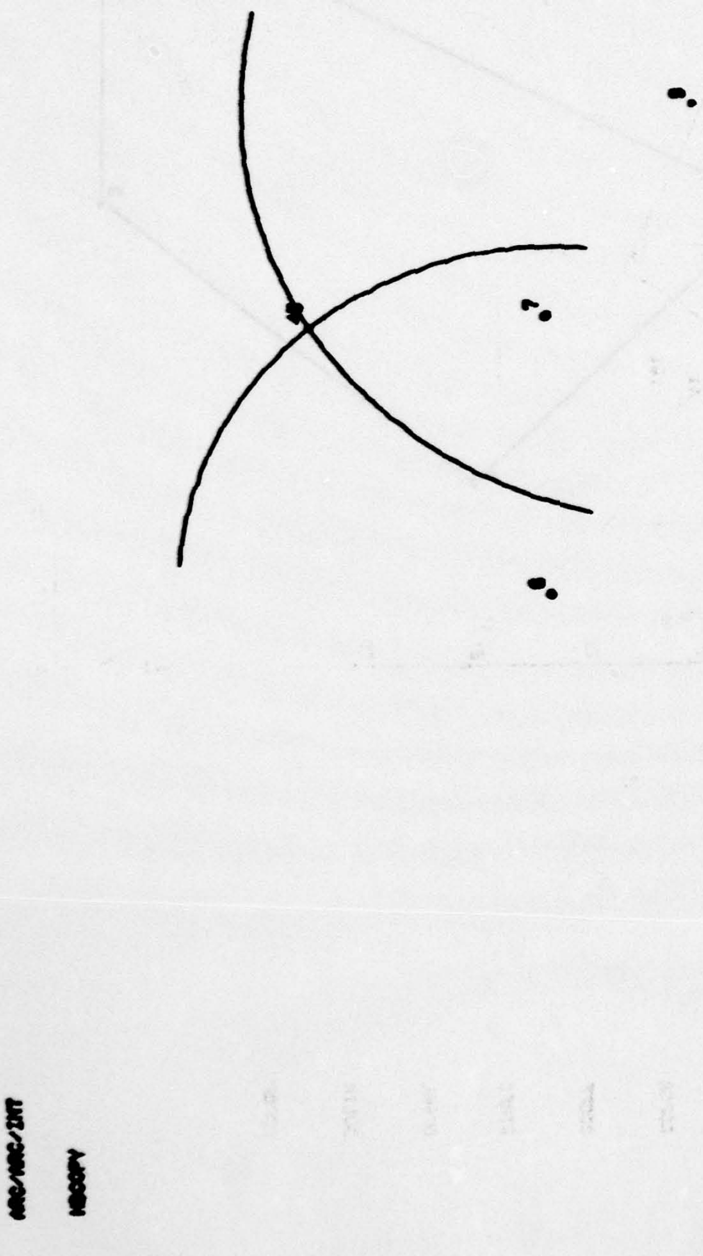
STAFC

OFSAL

SWLIN

HDCOPY

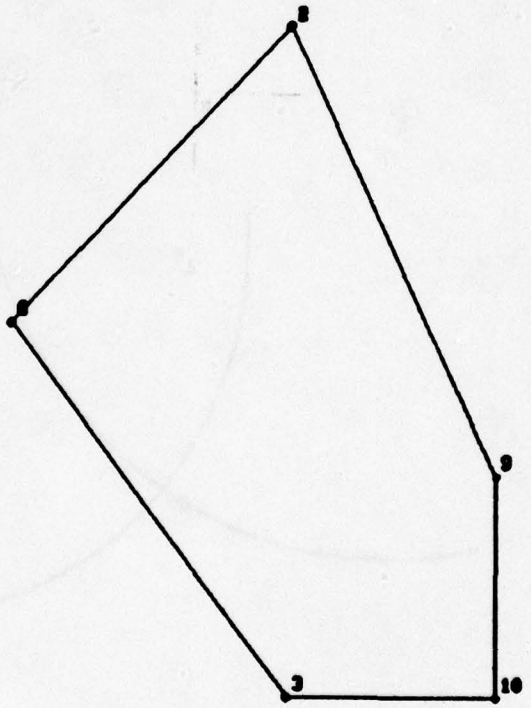



ARC/ARC/INT
HDCOPY

AREA/AZIMUTH

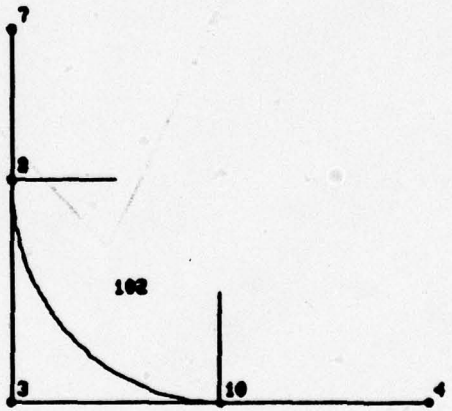
89888.9888 SQFT. 2.0661 ACRES

HBCOPY



GETCRV

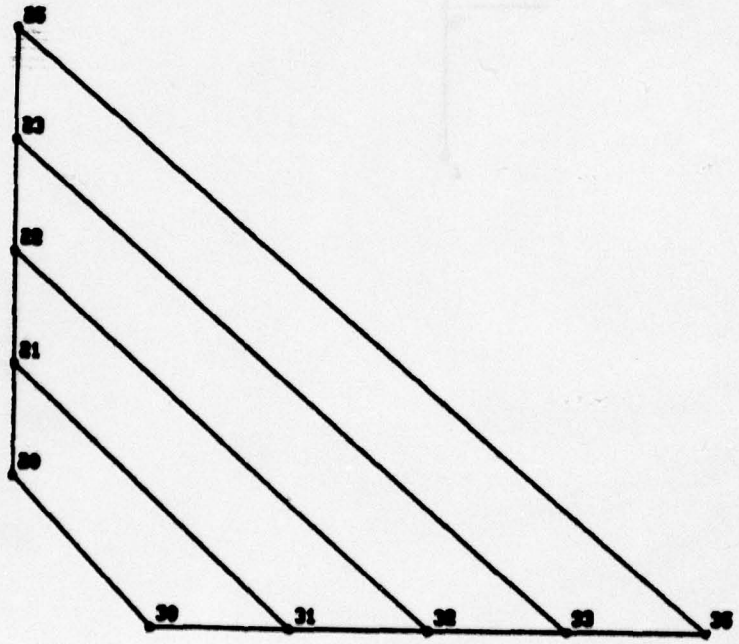
HDCOPY



SVLIN

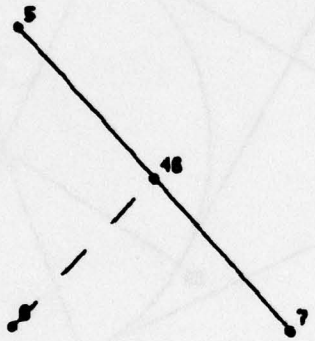
GIRSL

HSCCPV



TANGENT/OFF

HDCOPY



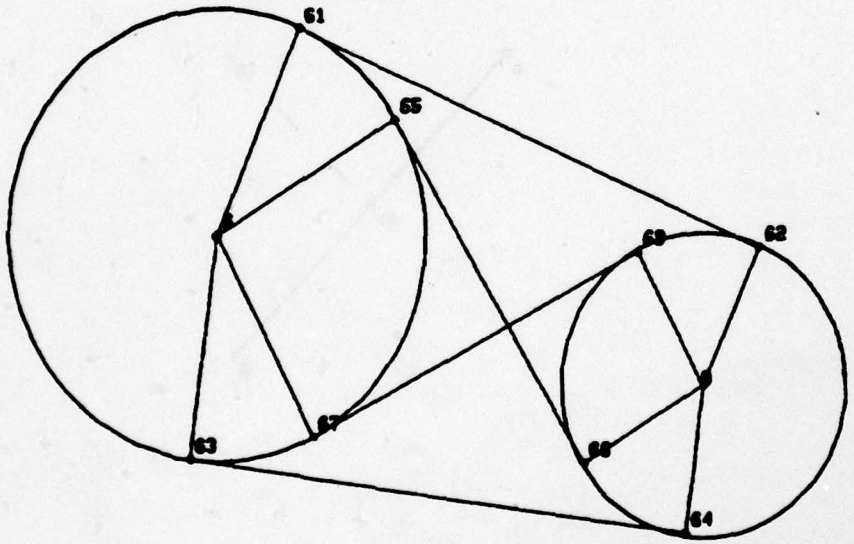
TANGENT

TANGENT

TANGENT

TANGENT

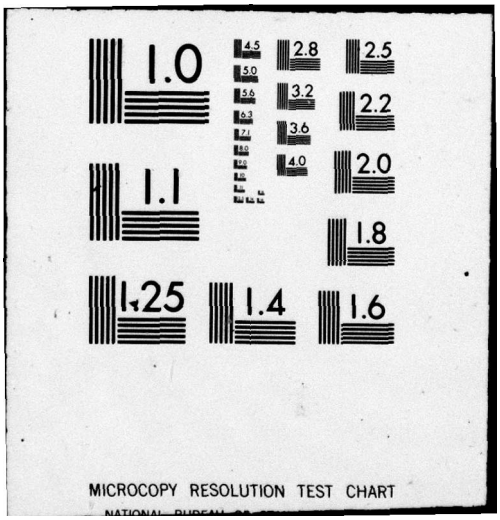
HDCOPY



AD-A082 128 ARMY ENGINEER WATERWAYS EXPERIMENT STATION VICKSBURG MS F/G 9/2
USER'S GUIDE: COMPUTER PROGRAM WITH INTERACTIVE GRAPHICS FOR CO--ETC(U)
FEB 80 H B WILSON, J L HILL
UNCLASSIFIED WES INSTRUCTION-K-80-2 NL

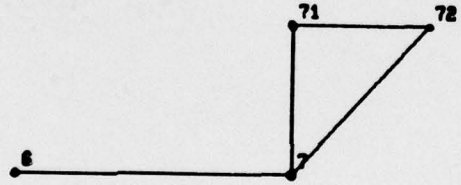
2 OF 2
ADA
082128

END
DATE
FILMED
4-80
DTIC


1.0
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.8
1.1
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART

TRIANGULATIO

HDCOPY

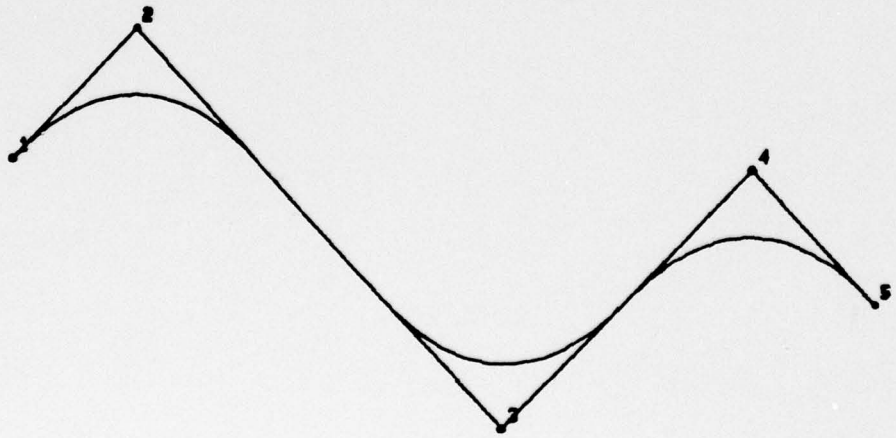


```
WINDOW

VERTICAL/STA

VERTICAL/END

HDCOPY
```



```
COMMENT
1050 EXAMPLE PROBLEM INVOLVING SEVERAL CURVES

CLEAR
BETWEEN POINTS     1  AND   500

STORE

DUMP

 POINT        Y              X
    2      4750.0000         0.
    4      2750.0000      1550.0000
    6      4000.0000      5750.0000
    8      2000.0000      3950.0000
   10      4950.0000     12350.0000
   12      8150.0000     11100.0000
   14      8100.0000      4000.0000
   15      3250.0000     11150.0000
   16      6250.0000      2250.0000

HDCOPY
```

WINDOW

ALIG
ALIGNMENT

ALIG
ALIGNMENT

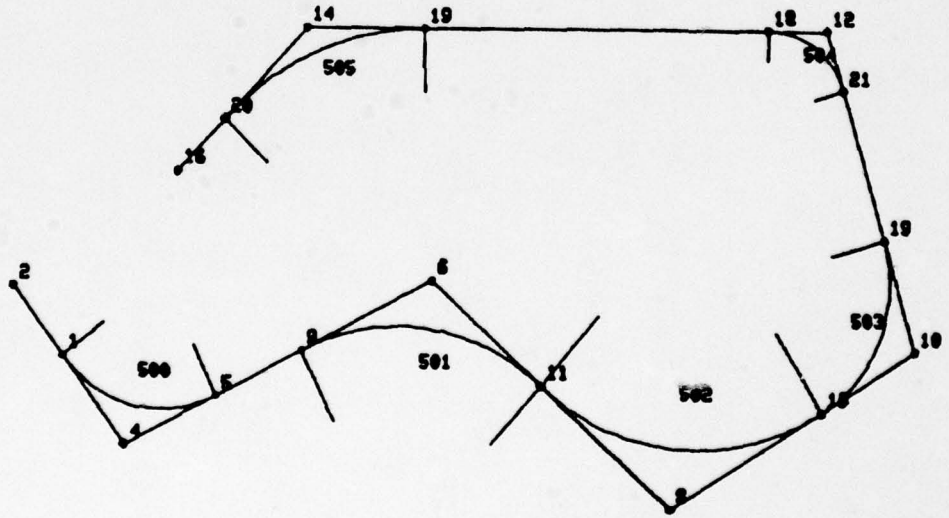
ALIG
ALIGNMENT

ALIG
ALIGNMENT

ALIG
ALIGNMENT

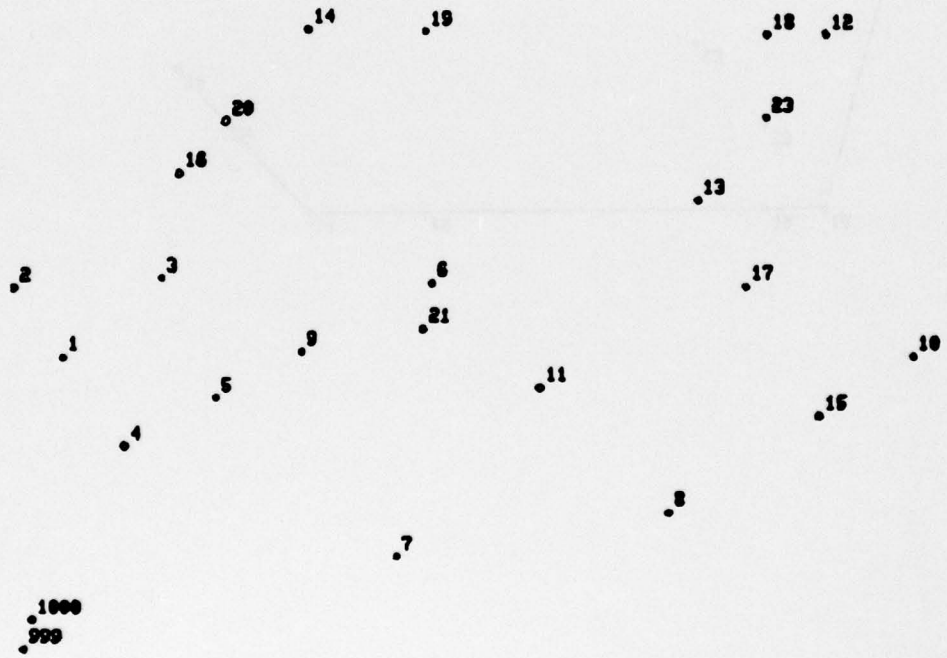
ALIG
ALIGNMENT

HDCOPY



SHOW/N-PTS

HDCOPY



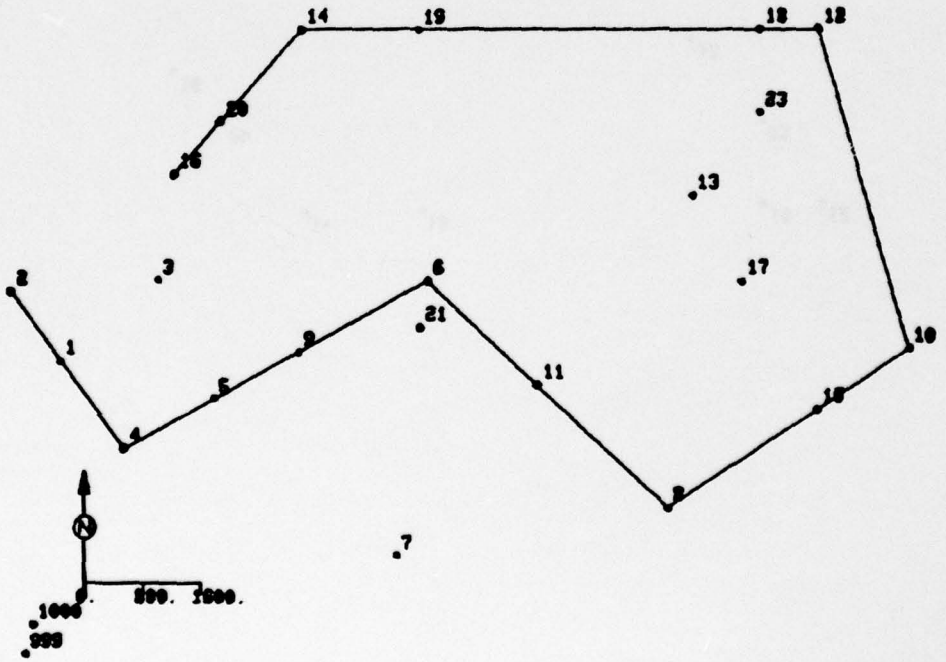
SHOW/N-PTS

HDCOPY

LINE/N-PTS

NORTH/SCALE

HDCOPY



SHOW/N-PTS

HDCOPY

LINE/N-PTS

NORTH/SCALE

HDCOPY

GETCRV

GETCRV

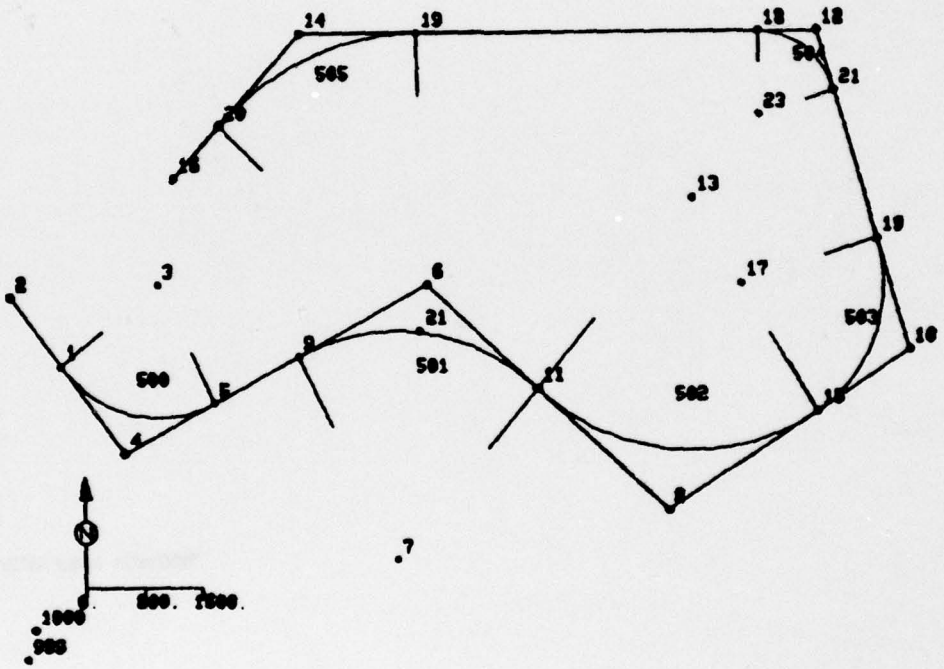
GETCRV

GETCRV

GETCRV

GETCRV

HDCOPY



```
COMMENT
1670 END OF DATAFILE HAS BEEN REACHED


PAUSE
  INPUT 0 TO READ FROM T.S. FILE OR 1 TO READ FROM TERMINAL
*1


NEXT COMMAND WORD
*END
DO YOU HAVE MORE DATA TO RUN?
(0 NO. 1 YES )
*0

E
```

In accordance with letter from DAEN-RDC, DAEN-ASI dated 22 July 1977, Subject: Facsimile Catalog Cards for Laboratory Technical Publications, a facsimile catalog card in Library of Congress MARC format is reproduced below.

Wilson, Howard B
User's guide: computer program with interactive graphics for coordinate geometry analysis / by Howard B. Wilson, James L. Hill, Systems Engineering Consultants, Inc., Tuscaloosa, Ala. Vicksburg, Miss. : U. S. Waterways Experiment Station ; Springfield, Va. : available from National Technical Information Service, 1980.
vi, 94 p. : ill. ; 27 cm. (Instruction report - U. S. Army Engineer Waterways Experiment Station ; K-80-2)
Prepared for U. S. Army Engineer Division, Lower Mississippi Valley, Vicksburg, Miss.

1. COGO. 2. Computer programs. 3. Coordinate geometry analysis. 4. Interactive computer graphics. I. Hill, James L., joint author. II. Systems Engineering Consultants, Inc. III. United States. Army. Corps of Engineers. Lower Mississippi Valley Division. IV. Series: United States. Waterways Experiment Station, Vicksburg, Miss. Instruction report ; K-80-2.
TA7.W34i no.K-80-2